BIBLIOTEKA NAUKOWA INŻYNIERA

Redaktorzy: MARCELI STARK, BOHDAN WALENTYNOWICZ

PAŃSTWOWE WYDAWNICTWO NAUKOWE

A. W. MOSTOWSKI i Z. PAWLAK

LOGIKA dla INŻYNIERÓW

WARSZAWA 1970

Redaktor odpowiedzialny
JANUSZ ONYSZKIEWICZ

Okładkę i obwolutę projektował
HENRYK BIAŁOSKÓRSKI

Redaktor
ROMANA EHRENFEUCHT

Redaktor techniczny
HANNA KĘSICKA

WROCŁAWSKA DRUKARNIA NAUKOWA

Przedmowa

Krąg osób zainteresowanych logiką objął w ostatnich czasach także i inżynierów. Konstrukcja maszyn matematycznych, automatyka i wiele innych działów techniki wymaga znajomości szeregu pojęć logicznych, a także umiejętności posługiwania się metodami używanymi w badaniu podstaw matematyki.

Na wyższych uczelniach technicznych logika na ogół nie jest wykładana. Tak więc technicy, z tytułu swojej pracy, „skazani" na znajomość logiki zmuszeni są uczyć się tego przedmiotu samodzielnie. Sprawa zdawałaby się całkiem prosta, gdyż zarówno w literaturze polskiej, jak i światowej jest wiele doskonałych podręczników z tego zakresu. Jednakże w samodzielnym studiowaniu tego działu matematyki występują dwie zasadnicze trudności.

Pierwsza trudność wynika z tego, że kurs matematyki na politechnice nie przygotowuje słuchaczy do studiowania zagadnień z dziedziny logiki. Stąd cały aparat pojęciowy logiki dla spotykającego się z nim po raz pierwszy inżyniera wydaje się niezrozumiały i sztuczny.

Druga trudność natomiast pochodzi stąd, że duża część materiału jest po prostu trudną do znalezienia, gdyż te pojęcia logiczne, które ostatnio znajdują zastosowanie w technice, są rzadko podawane w podręcznikach logiki; na przykład różne pojęcia algorytmów, języka maszyny itp.

Celem przedstawionej Czytelnikowi książki jest pomoc w pokonaniu tych dwóch trudności. Po pierwsze, staraliśmy się podać materiał w prostej i przystępnej postaci, omawiając możliwie szczegółowo te sprawy, które dla matematyka mogą się wydawać proste i oczywiste, a nawet nieciekawe.

Po drugie, staraliśmy się uwzględnić te wszystkie działy logiki i dziedzin pokrewnych, które naszym zdaniem są potrzebne inżynierowi.

Nie łudzimy się, że zadanie to udało się nam w pełni zrealizować, niemniej jednak sądzimy, że uwagi uzyskane od Czytelników pozwolą nam w przyszłości lepiej zrealizować powyższy program.

Na zakończenie tej krótkiej przedmowy chcielibyśmy wyrazić podziękowanie doc. M. Starkowi za propozycję napisania tej książki. Prof. H. Rasiowej pragniemy wyrazić naszą głęboką wdzięczność za szereg cennych uwag odnośnie treści, jak i redakcji książki oraz jej terminologii.

Chcielibyśmy również podziękować drowi L. Szczerbie, który był de facto pierwszym Czytelnikiem tej książki przed złożeniem jej w redakcji.

Autorzy

Warszawa, 1969

Rozdział 1

WSTĘP

Logika zajmuje się prawami myślenia — wyciągania wniosków z sądów w taki sposób, by były one poprawne. Jednak tylko niewielki fragment książki, którą chcemy przedstawić zajmuje się prawami wnioskowania (rozdziały II i III).

Całość książki poświęcona jest różnym teoriom i zagadnieniom matematycznym, bądź też im pokrewnym, wynikłym z badań nad logiką. Rozdziały V i VI poświęcone są podstawowym pojęciom matematycznym — zbiorom i relacjom. Ogólny opis najprostszych teorii matematycznych, tzw. teorii elementarnych, podany jest w rozdziale IV. Rozdziały VII i VIII poświęcone są algebrom Boole'a i półgrupom ważnym z uwagi na zastosowania w badaniach nad automatami i maszynami matematycznymi. Rozdziały te, jak również częściowo rozdziały V i VI stanowią ponadto ilustracje pojęć zawartych w rozdziale IV, poświęconym teoriom sformalizowanym.

Rozdziały IX-XII poświęcone są omówieniu ważnego zagadnienia związanego z budową teorii matematycznych w sposób sformalizowany — zagadnienia rozstrzygalności teorii. Studia nad tym zagadnieniem doprowadziły do wykształcenia się pojęć mających obecnie chyba największe znaczenie dla badań nad maszynami matematycznymi i urządzeniami automatycznymi. Pojęcia te to pojęcie algorytmu (rozdział IX) oraz funkcji rekurencyjnych (rozdział XII).

Książka nie jest podręcznikiem logiki i nie zajmuje się studiami nad nią. Celem jej jest przedstawienie tych działów związanych z logiką, których znajomość jest użyteczna w pracy inżyniera zajmującego się teorią maszyn matematycznych, urządzeń automatycznych, zagadnieniami języków maszynowych i wieloma innymi. Pominięte zostały w niej zagadnie-

nia pokrewne, takie jak np. teoria automatów skończonych, teoria gier i zagadnienia probabilistyczne związane z teorią informacji. Zagadnienia te są zbyt odległe od poruszanych przez nas problemów, a ponadto mają oddzielne monografie i opracowania.

Książka choć zajmuje się teoriami sformalizowanymi, jednak, ze względów utylitarnych, na ogół nie używa języka sformalizowanego. Zagadnienie formalizacji poruszane jest w książce pod kątem zagadnień praktycznych. Kwestie przeprowadzenia formalizacji matematyki, jak również związane z tym zagadnienia filozoficzne odsunięte są na plan dalszy.

Rozdział początkowy książki, który przedstawiamy poniżej, poświęcony jest ogólnemu scharakteryzowaniu tych zagadnień. W paragrafie 1 i 2 wyjaśnione jest co to są teorie dedukcyjne, zwane inaczej teoriami aksjomatycznymi. W paragrafie 3 podane są przyczyny, dla których budowa matematyki musiała osiągnąć wyższy stopień ścisłości — formalizację. W paragrafie 4 omówiony jest problem, do jakiego stopnia bogatą rzeczywistość matematyczną można przedstawić za pomocą jej sformalizowanego opisu. Ostatni paragraf 5 wstępu przedstawia zagadnienia rozstrzygalności i różne metody ujmowania tego zagadnienia.

Cały rozdział wstępny odbiega nieco swoim charakterem od reszty książki. Używa się w nim języka potocznego, intuicyjnie zrozumiałego i raczej opowiada się o sprawie formalizacji niż ją definiuje.

§ 1. O TEORIACH DEDUKCYJNYCH

Każda nauka podaje interesujące ją fakty w postaci zdań zwanych *tezami*, bądź *twierdzeniami* tej nauki. Zbiór takich zdań nazywany jest *teorią*. Każda teoria jest więc zbiorem zdań oznajmujących, to znaczy zdań wyrażających jakieś sądy.

Twierdzenia teorii muszą być oczywiście zdaniami prawdziwymi, tj. muszą wyrażać sądy zgodne z rzeczywistością, inaczej teoria pozbawiona byłaby jakiejkolwiek wartości poznawczej i praktycznej. Tak więc każda teoria jest zbiorem pewnych zdań prawdziwych.

Wśród teorii naukowych szczególną pozycję zajmują teorie matematyczne. Każda teoria matematyczna jest oczywiście również zbiorem zdań prawdziwych (twierdzeń). Ponadto zdania teorii matematycznych charakteryzują się szczególnym rodzajem wzajemnego powiązania twierdzeń — cechą, która nie musi przysługiwać innym teoriom. Mianowicie wszystkie twierdzenia w dowolnej teorii matematycznej otrzymujemy drogą logicznych rozumowań z niewielkiej liczby twierdzeń wyjściowych zwanych *aksjomatami* lub *pewnikami*. Teorie o takiej budowie są nazywane *teoriami dedukcyjnymi* lub *teoriami aksjomatycznymi*.

Twierdzeniami teorii dedukcyjnej są więc te zdania, które są albo aksjomatami teorii, albo też dadzą się otrzymać z aksjomatów drogą logicznego rozumowania.

Prawa logicznego rozumowania ujęte są w pewną ilość prostych reguł, pozwalających ze zdań prawdziwych zwanych *przesłankami* otrzymywać nowe zdania prawdziwe — *wnioski*. Z regułami tymi zapoznamy się w rozdziałach II i III. Ich cechą charakterystyczną jest to, że o przyjęciu wniosku nie decydujemy na podstawie jego treści czy też treści przesłanek, a na podstawie ich postaci.

Do określenia teorii dedukcyjnej nie wystarcza wyliczenie aksjomatów. Konieczne jest sprecyzowanie praw wnioskowania, z których będziemy korzystali. Prawa wnioskowania, z których korzystamy w danej teorii dedukcyjnej noszą nazwę *reguł dedukcji* tej teorii.

Streszczenie

Teorię nazywamy teorią dedukcyjną lub aksjomatyczną, jeżeli spośród twierdzeń teorii wyróżniony jest zbiór zdań zwanych aksjomatami lub pewnikami oraz podany jest zbiór reguł dedukcji (reguł wnioskowania) pozwalających z dowolnych twierdzeń danej teorii uzyskiwać inne twierdzenia tej teorii.

§ 2. ZNACZENIE TEORII DEDUKCYJNYCH

W tym paragrafie są przedstawione zastosowania teorii dedukcyjnych.

Zastosowanie jakiejś teorii dedukcyjnej polega na korzystaniu z twierdzeń tej teorii w innych gałęziach nauk. Odpowiedź na pytanie: w jakim stopniu otrzymane tą drogą wyniki odpowiadają rzeczywistości, sprowadza się właściwie do sprawdzenia, czy użyta teoria dobrze tę rzeczywistość

opisuje. Jeżeli tak, nie ma powodu, aby otrzymane w drodze logicznego rozumowania wnioski były niezgodne z rzeczywistością. W ten sposób np. matematyka pozwala przewidywać nowe fakty, bez konieczności uciekania się do eksperymentu. Ten schemat stosowania matematyki ukształtował się już w XIX wieku, głównie na bazie analizy matematycznej i rachunku prawdopodobieństwa.

Matematycy patrzą jednak nieco inaczej na teorie dedukcyjne — z punktu widzenia własnej gałęzi wiedzy. Teorie dedukcyjne są nie tylko narzędziem przewidywania nowych faktów. Znaczenie ich jest głębsze.

Wypowiedzi będące twierdzeniami teorii zawierają pewne pojęcia. Jedne z nich dają się określić przez inne, ale pozostaje pewna liczba pojęć, których przez inne określić się nie da, choć same one pozwalają określić wszystkie pojęcia występujące w teorii. Są to tak zwane *pojęcia pierwotne* teorii. W mechanice pojęciami takimi są np. masa, długość i czas. W geometrii płaskiej pojęciami pierwotnymi są np. pojęcie punktu, prostej, leżenie punktu na prostej, leżenie punktu na prostej między dwoma innymi punktami oraz przystawaniem figur.

Pojęcia pierwotne nie są określone przez inne pojęcia. Są one scharakteryzowane przez wybór aksjomatów. Wyjaśnimy to pokrótce. W zastosowaniach teorii dedukcyjnej interpretujemy w pewien sposób wszystkie twierdzenia teorii. Interpretacja ta odbywa się przez przypisanie jakiegoś znaczenia pojęciom pierwotnym. Wybór tego znaczenia nie jest dowolny. Ograniczeni jesteśmy żądaniem, aby przy tej interpretacji aksjomaty były wypowiedziami prawdziwymi. Aksjomaty określają więc znaczenie pojęć pierwotnych.

W geometrii płaskiej punkty możemy interpretować jako pary liczb rzeczywistych, proste jako równania liniowe itd. Tak zinterpretowana na gruncie teorii liczb rzeczywistych geometria nosi nazwę geometrii analitycznej.

Geometria płaska (mowa o geometrii euklidesowej) jest ciekawym przykładem *teorii* tzw. *kategorycznej*, której pojęcia pierwotne można interpretować właściwie tylko na jeden sposób. Każde dwie interpretacje będą się różnić od siebie nieistotnie. Teorie kategoryczne, aczkolwiek ich twierdzenia mają duże znaczenie dla nauki, są właściwie nieciekawe. Jednak są to jedyne teorie, które w zupełności charakteryzują pojęcia pierwotne. Najciekawsze, z uwagi na zastosowanie, są *teorie niekategoryczne* dopuszczające wiele istotnie różnych interpretacji i często w bardzo odległych dziedzinach wiedzy. Przykładem takiej teorii jest np. teoria algebr Boole'a opisana w rozdziale VII. Czytelnikowi zapewne wiadomo, że twierdzenia tej algebry można interpretować zarówno w logice, jak i teorii zbiorów jak również w dziedzinie pozornie odległej od nich — w teorii sieci przekaźnikowych.

Teorie o wielorakich interpretacjach budzą ostatnio coraz większe zainteresowanie. Jedna i ta sama teoria, dzięki interpretacjom w różnych gałęziach wiedzy, może interesować nie tylko matematyków (logików) czy fizyków lecz również inżynierów zajmujących się teorią sieci elektrycznych i maszyn matematycznych, biologów, lingwistów, ekonomistów, czy wreszcie prawników.

Streszczenie

Omówiliśmy pokrótce sprawę zastosowań teorii dedukcyjnych i ich znaczenia. Teorie dedukcyjne przez swoje aksjomaty opisują własności pojęć pierwotnych występujących w teorii. Z twierdzeń wynikają nowe fakty. Twierdzenia te pozwalają stosować wyniki teorii w praktyce. Teoria dedukcyjna stworzona w jednym dziale nauki może być drogą interpretacji wykorzystywana w innym dziale. W zależności od interpretacji jej pojęć pierwotnych może być więc teorią wspólną dla kilku nauk.

§ 3. TEORIE SFORMALIZOWANE

Język i pojęcia używane przez matematyków, choć od wielu stuleci wydawały się szczytem precyzji i dokładności, okazywały się często tak niedokładne, że prowadziły do paradoksów.

Rozpatrzmy następujący przykład pokazujący do jakich niebezpieczeństw prowadzi posługiwanie się pojęciami intuicyjnymi.

Wydawałoby się, że mając jakąś własność można by mówić o zbiorze elementów, które tę własność posiadają. Weźmy na przykład następującą własność W dotyczącą zbiorów: Zbiór X ma własność W, jeśli nie jest swoim własnym elementem.

Rozpatrzmy zbiór, którego elementami są te i tylko te zbiory, które nie są swoimi własnymi elementami, czyli mają własność W, i spróbujmy stwierdzić, czy zbiór ten jest swoim własnym elementem, czy też nie. Przypuśćmy, że jest on swoim własnym elementem. Otrzymujemy sprzeczność z definicją tego zbioru, gdyż on sam jako swój własny element musi (z definicji) nie być swoim własnym elementem. Przypuśćmy więc, że zbiór ten nie jest swoim własnym elementem. Wtedy zgodnie z definicją jest jednym ze swoich elementów. Obaliliśmy więc oba przypuszczenia, zarówno to, że określony przez nas zbiór jest swoim własnym elementem jak i przeciwnie.

Powyższe rozumowanie wydaje się zupełnie poprawne. Przesłanka, że dla dowolnych elementów można utworzyć zbiór złożony z tych i tylko tych elementów, które mają jakąś własność, wydaje się nie budzić wątpliwości i intuicyjnie wydaje się być prawdziwa, a jednak wniosek z niej otrzymany jest nie do przyjęcia, gdyż jest fałszywy.

Jednak nie tylko sądy wydające się być intuicyjnie prawdziwe doprowadzały do paradoksów i sprzeczności. Źródła sprzeczności mogły tkwić i w samym języku. Do ilustracji tego pomocny nam będzie następujący przykład:

Określenie „najmniejsza liczba naturalna, której nazwy nie można zapisać za pomocą mniej niż stu znaków pisarskich" na pierwszy rzut oka nie budzi wątpliwości. Zauważmy jednak, że w powyższym zdaniu określiliśmy tę liczbę pisząc mniej niż sto znaków pisarskich. Źródłem sprzeczności był tu niezbyt precyzyjny język, którego używaliśmy.

Podobnego typu sprzeczności zwanych *antynomiami* odkryto do końca dziewiętnastego wieku wiele. Pojawienie się ich pokazało, że logika używana w sposób intuicyjny, intuicyjne używanie pojęć matematycznych, wypowiadanie sensu twierdzeń matematycznych w sposób intuicyjny niesie w sobie wiele niebezpieczeństw.

Aksjomatyczna budowa matematyki zabezpieczyła ją od wielu paradoksów. Pozostały jednak te, których źródłem okazał się nie dość ściśle sprecyzowany język matematyki oraz dowolność w wyborze reguł wnioskowania. Określając system dedukcyjny nie podaliśmy wyraźnie, jakie reguły logiczne wolno nam stosować w rozumowaniach matematycznych, a jakie nie. Nie precyzowaliśmy też w jaki sposób wyrażamy twierdzenia teorii. Dla uniknięcia paradoksów okazało się konieczne sprecyzowanie ścisłe reguł wnioskowania, którymi wolno się nam posługiwać w dowo-

dzeniu twierdzeń, jak również uściślenie języka, w którym wypowiadamy twierdzenia teorii.

Teoria dedukcyjna operująca ściśle sprecyzowanym językiem oraz wyraźnie sprecyzowanymi regułami dedukcji nazywa się *teorią sformalizowaną*.

Na przełomie dziewiętnastego i dwudziestego wieku podwaliny matematyki: logika, teoria zbiorów i arytmetyka zostały ujęte w ramy sformalizowanych teorii. Pokazano, że w precyzyjnie określonym języku, przy ścisłym podaniu reguł wnioskowania i starannym doborze aksjomatów, żaden z dotychczas znanych paradoksów nie może być wysłowiony. Przeprowadzenie dowodu zostało sprowadzone do pewnego postępowania wychodzącego z aksjomatów i polegającego na stosowaniu reguł wnioskowania. Zostało ono w pewien sposób „zmechanizowane". Intuicja została zastąpiona ścisłymi przepisami co wolno robić, a co nie.

Powstało pytanie, czy dzięki temu matematyka zyskała i czy nie została w ten sposób zubożona.

Streszczenie

Stwierdziliśmy, że potrzeba formalizacji matematyki powstała z powodów wewnętrznych. Należało właściwie sprecyzować o czym się w matematyce mówi i jakie aksjomaty i środki dowodzenia należy przyjąć, by uniknąć paradoksów powstających przy użyciu intuicyjnego języka i intuicyjnych pojęć.

§ 4. FORMALIZACJA MATEMATYKI

W paragrafie tym zajmiemy się kwestią, jak dalece aksjomaty i reguły wnioskowania opisują matematykę.

Reguły wnioskowania używane w sformalizowanych teoriach matematycznych są tak dobrane, by ze zdań prawdziwych otrzymywać zdania prawdziwe. Aksjomaty sformalizowanych teorii matematycznych opisują podstawowe własności pojęć pierwotnych teorii. Na przykład aksjomatyka liczb naturalnych opisuje podstawowe własności liczb naturalnych. Dobrze by było, aby przy odpowiednim doborze aksjomatów i reguł wnioskowania były teorie, które w pełni opisują różne działy matematyki.

Twierdzenia pewnej teorii są zdaniami prawdziwymi. Czy wszystkie zdania dające się wypowiedzieć w jakiejś teorii, które uważamy za prawdziwe na podstawie rozumienia ich sensu dadzą się otrzymać z aksjomatów dzięki stosowaniu do nich reguł dedukcji — inaczej mówiąc czy wszystkie zdania prawdziwe teorii będą twierdzeniami teorii?

Badania nad matematycznymi teoriami sformalizowanymi wykazały, że na ogół tak nie jest. W większości teorii istnieją zdania, które są prawdziwe, ale nie są twierdzeniami teorii. Co więcej rzeczywistość matematyczna jest tak bogata, że nie można podać explicite zbioru aksjomatów i reguł dedukcji dających się stosować w sposób mechaniczny, tak by zakres twierdzeń pokrywał się z zakresem zdań prawdziwych. Sytuacja w fizyce jest dość podobna; żadna teoria fizyczna nie przewiduje wszystkich faktów doświadczalnych.

Formalizacja całej matematyki okazała się programem nie do przeprowadzenia. Badania nad nią pogłębiły jednak wiedzę o matematyce i umożliwiły zastosowania sformalizowanych teorii matematycznych w innych działach nauki. Badania te doprowadziły do budowy, na wzór teorii sformalizowanych, innych teorii, w których wyrażenia nie są interpretowane jako sądy, reguły dedukcji zaś nie są wynikiem praw wnioskowania logicznego. Teorie takie nie służą już oczywiście formalizacji matematyki i opisują zupełnie inną rzeczywistość niż matematykę. Na przykład usiłuje się ostatnio tworzyć i badać sformalizowane teorie lingwistyczne, biologiczne, sformalizowane teorie urządzeń automatycznych i inne.

Badania nad tymi teoriami, choć korzystają ze zdobyczy matematyki mają zupełnie inny cel i zakres (celem ich nie jest tak jak dla teorii logicznych i matematycznych uzyskanie możliwości badania prawdy i fałszu, lecz przede wszystkim opis innych aspektów badanej rzeczywistości, uzyskiwanie nowych faktów).

Streszczenie

Stwierdziliśmy, że formalizacja matematyki nie daje pełnego opisu zdań prawdziwych odnoszących się do rzeczywistości matematycznej. Badania teorii sformalizowanych mają jednak duże znaczenie dzięki nowym ich zastosowaniom.

§ 5. ROZSTRZYGALNOŚĆ TEORII SFORMALIZOWANYCH

Studia nad rozstrzygalnością teorii stanowią ważną i znajdującą obecnie może najgłębsze zastosowanie w teorii maszyn matematycznych dziedzinę badań nad teoriami sformalizowanymi.

Kwestię rozstrzygalności teorii formułuje się następująco.

Czy istnieje jakaś efektywna metoda pozwalająca stwierdzić w skończonej liczbie kroków, czy dana dowolna formuła jest twierdzeniem teorii, czy też nie? Przez metodę rozstrzygania jakiegoś zagadnienia rozumiemy przepis pozwalający efektywnie w sposób mechaniczny rozstrzygnąć to zagadnienie w każdym konkretnym przypadku, bez potrzeby rozumienia wykonywanych czynności oraz ich treści matematycznej. Oczywiście takie podejście jest bardzo nieprecyzyjne. Dlatego matematycy badający kwestie rozstrzygalności teorii starali się precyzyjnie zdefiniować co to znaczy, że istnieje metoda pozwalająca coś rozstrzygnąć.

Powstały różne definicje „metody" oparte na różnych intuicjach związanych z tym słowem. Wszystkie one zawierały pewną cechę wspólną odpsychologizowania pojęcia metody — tak by tryb wykonywania i postępowania wyznaczonego przez „metodę" był efektywnie wykonalny niezależnie od umiejętności matematycznych przeprowadzającego operację.

Intuicje, które posłużyły do uściślenia pojęcia metody, szły w trzech kierunkach.

Pierwszy z nich polegał na uściśleniu pojęcia przepisu, przez podanie z jakich reguł postępowania mogą składać się te przepisy oraz jak należy podane reguły stosować, tak aby otrzymać interesujący nas wynik. W ten sposób powstało pojęcie algorytmu, a ściśle nie jedno, a wiele pojęć algorytmów.

Drugi kierunek nawiązywał do pojęcia maszyny. Istnieje metoda rozstrzygania, gdy można podać maszynę, która po dostarczeniu jej formuły wykona odpowiednie czynności i zasygnalizuje odpowiedź, czy podana formuła jest twierdzeniem, czy też nie. Nie chodzi tu oczywiście o podanie technicznej konstrukcji maszyny, tzn. maszyny, którą można naprawdę zbudować, a o pewną koncepcję teoretyczną maszyny przez podanie jej opisu działania.

Trzeci wreszcie kierunek, pojęcie metody efektywnej sprowadził do elementarnych operacji arytmetycznych na liczbach naturalnych, takich jak

dodawanie, mnożenie itp., tak że zamiast rozpatrywać formuły jakiejś teorii sformalizowanej, rozważa się odpowiednio im przyporządkowane liczby naturalne. W ten sposób problem rozstrzygalności można sformułować na gruncie arytmetyki. Ponieważ wykonywanie operacji arytmetycznych jest czynnością ściśle sprecyzowaną, pojęcie metody rozstrzygania zostało tym samym również uściślone. Z takiego uściślenia powstał dział matematyki liczb naturalnych zwany teorią funkcji rekurencyjnych.

Teoria funkcji rekurencyjnych prowadzi do najbardziej zamkniętego i zwartego w sobie pojęcia metody rozstrzygania jakiejś klasy zagadnień. Badania nad algorytmami i maszynami nie są natomiast jeszcze zakończone — brak jest ogólnej syntezy tych zagadnień. Co jednak poza pasją naukową skłania matematyków do badania tych pojęć?

Wydaje się, że przyczyną tą jest fakt, że zagadnienia związane z tymi pojęciami wyszły daleko poza ich zastosowania do kwestii rozstrzygalności różnych zagadnień matematycznych. Rozwój cyfrowych maszyn matematycznych pokazał nowe zastosowania i użyteczność tych pojęć stworzonych dla celów teoretyczno-matematycznych (kwestie te omówione są w rozdziale XI i XII). Zagadnienia te interesują już nie tylko matematyków. Obecnie musi je znać również inżynier zajmujący się teorią cyfrowych maszyn matematycznych, ale nie tylko on. Wyniki dotyczące algorytmów i maszyn matematycznych znalazły zastosowania również np. w lingwistyce matematycznej (por. rozdział IX paragraf 52), czy też teorii sterowania i programowania czynności. Problem zastosowania tych wyników w innych działach nauki jest otwarty. Dopiero przyszłość może pokazać ich pełną użyteczność.

Streszczenie

Omówiliśmy trzy sposoby uściślenia pojęcia metody rozstrzygania jakiegoś zagadnienia: 1. pojęcie algorytmu; 2. pojęcie maszyny; 3. pojęcie funkcji rekurencyjnej.

Pojęcia te zostały stworzone przez matematyków w związku z rozwojem teorii sformalizowanych, jednak ich znaczenie i zastosowanie wychodzą obecnie daleko poza rozumowania sensu stricto matematycznego.

Rozdział 2

RACHUNEK ZDAŃ

Wykład logiki — nauki o poprawnych formach wnioskowania — tradycyjnie jest zaczynany od podstawowego jej fragmentu, rachunku zdań. Wiemy z części pierwszej, że każda teoria matematyczna jest zbiorem zdań i że zdania takiej teorii są jakoś ze sobą powiązane; między zdaniami teorii zachodzą pewne związki. Zdanie jest podstawową cegiełką teorii matematycznych, nic więc dziwnego, że jest ono pierwszym pojęciem, które należy zbadać i wyjaśnić przed przystąpieniem do jakichkolwiek rozważań na temat teorii matematycznych.

Badanie zdań wchodzących w skład teorii matematycznych i zachodzących między nimi związków doprowadziło do powstania nowej teorii matematycznej — rachunku zdań, który mamy zamiar przedstawić w tym rozdziale. Bardziej szczegółowe wiadomości na ten temat znajdzie Czytelnik w każdym podręczniku logiki (patrz np. Mostowski, Grzegorczyk [1955], Grzegorczyk [1961], Kleene [1952], Nowikow [1959], Rasiowa [1966] i [1968], Pogorzelski i Słupecki).

Czytelnik spotykający się po raz pierwszy z pojęciami rachunku zdań może czuć się nieco zdziwiony, jak bardzo pojęcia te odbiegają od pojęć gramatycznych. Zakresy rachunku zdań i gramatyki tylko częściowo się zazębiają, jednakże w istocie problematyka obu tych dziedzin jest różna. Głównym zadaniem gramatyki jest badanie reguł pozwalających odróżniać zdania poprawne od zdań niepoprawnie zbudowanych w języku naturalnym (polskim, angielskim czy japońskim). Natomiast głównym celem rachunku zdań jest badanie prawdziwości zdań zależnie od ich struktury i to nie wszystkich zdań, a zdań którym można przypisać wartość prawdy lub fałszu. Zagadnienia poprawności struktury zdań wchodzą oczywiście również częściowo w zakres rachunku zdań, jednakże w znacznie mniejszym stopniu aniżeli ma to miejsce w gramatyce.

§ 6. SPÓJNIKI ZDANIOWE

Główną rolę w teoriach matematycznych grają zdania złożone ze zdań prostych za pomocą spójników takich jak:

... lub ...
... i ...
jeżeli ..., to ...
... wtedy i tylko wtedy, gdy ...
nie ...

W języku potocznym powyższe spójniki grają również ważną rolę, chociaż może nie tak dużą jak w rachunku zdań. Wystarczy wziąć jakikolwiek podręcznik matematyki, aby stwierdzić, że niemal w każdym zdaniu występują one wielokrotnie, natomiast w tekście niematematycznym spotyka się je znacznie rzadziej(1).

Łącząc jakiekolwiek dwa zdania za pomocą jednego z pierwszych czterech spójników otrzymamy nowe zdanie poprawne. Na przykład jako zdania proste weźmy zdania:

(1) 10 jest liczbą parzystą,

(2) 10 jest liczbą nieparzystą.

Łącząc oba powyższe zdania spójnikiem „lub", otrzymamy nowe zdanie

(3) 10 jest liczbą parzystą *lub* 10 jest liczbą nieparzystą.

Zdanie otrzymane przez połączenie dwu (lub więcej) zdań spójnikiem „lub" nazywamy *alternatywą* lub też *sumą logiczną* zdań składowych(2). Zdanie (3) jest więc sumą logiczną zdań (1) i (2).

Podobnie łącząc zdania proste

(4) 10 jest liczbą parzystą,

(5) 10 jest większe od 2

(1) Przez tekst matematyczny rozumiemy tutaj twierdzenia, definicje itp., tj. zdania jakiejś teorii matematycznej, a nie objaśnienia, czy też zdania wiążące w całość poszczególne twierdzenia. Zdania te należą oczywiście do języka potocznego.

(2) W literaturze rosyjskiej alternatywę nazywa się często *dyzjunkcją*, jednak w polskiej terminologii matematycznej termin dyzjunkcja ma inne znaczenie.

spójnikiem „i", otrzymamy zdanie złożone

(6) 10 jest liczbą parzystą *i* 10 jest większe od 2,

które jest nazywane *koniunkcją* albo *iloczynem logicznym* zdań składowych. Zdanie (6) jest więc iloczynem zdań (4) i (5).

Jeżeli zdania

(7) 10 jest liczbą podzielną przez 2,

(8) 10 jest liczbą parzystą

połączymy spójnikiem „jeżeli ..., to", to otrzymamy zdanie złożone następującej treści:

(9) *Jeżeli* 10 jest liczbą podzielną przez 2, *to* 10 jest liczbą parzystą.

Tak otrzymane zdanie jest nazywane *implikacją* zdań składowych(1).

Ten typ zdań jest bodaj najczęściej spotykany w matematyce. Większość twierdzeń matematycznych ma właśnie postać implikacji.

Również zwrot „wtedy i tylko wtedy, gdy" jest bardzo często używany w matematyce. Na przykład posługując się tym spójnikiem, ze zdań (7) i (8) otrzymamy zdanie:

(10) 10 jest liczbą podzielną przez 2 *wtedy i tylko wtedy, gdy* 10 jest liczbą parzystą.

Zdanie uzyskane przez połączenie dwu zdań zwrotem „wtedy i tylko wtedy, gdy" jest nazywane *równoważnością*.

Wreszcie ostatni zwrot „nie" napisany przed jakimkolwiek zdaniem tworzy wraz z nim nowe zdanie zwane jego *zaprzeczeniem* albo *negacją*. Na przykład pisząc przed zdaniem

5 jest liczbą podzielną przez 2

zwrot „nie" otrzymamy zdanie:

Nie, 5 jest liczbą podzielną przez 2.

(1) Nazwanie zwrotu „jeżeli..., to" spójnikiem może budzić wątpliwości. Zwróćmy jednakże uwagę, że z gramatycznego punktu widzenia rola tego zwrotu jest podobna do spójników „i" oraz „lub", gdyż z dwu zdań składowych pozwala on stworzyć nowe zdanie. W tym sensie nazwanie tego zwrotu spójnikiem jest uzasadnione.

Na to, by być w zgodzie z duchem gramatycznym języka, zdanie to czytamy czasami w następujący sposób: „Nieprawda, że 5 jest liczbą podzielną przez dwa". Nie jest to jednak całkowicie zgodne ze znaczeniem spójnika „nie", gdyż zwrot „nie" oznacza zaprzeczenie, a zwrot „nieprawda, że" ma charakter wartościujący sąd.

Zwrot „nie" nazwaliśmy również spójnikiem zdaniowym, chociaż odnosi się on tylko do jednego zdania. Spójniki międzyzdaniowe pozwalają bowiem tworzyć z jednych zdań nowe zdania. Zamiast terminu spójnik zdaniowy, który jest używany raczej w gramatyce, w logice stosuje się często zwrot: *funktor zdaniotwórczy*. W tym paragrafie pozostaniemy jednak przy pierwszym terminie.

Na zakończenie paragrafu o spójnikach zdaniowych, zwrócimy uwagę jeszcze na fakt, który dla osób stykających się po raz pierwszy z rachunkiem zdań wydaje się dziwny. Mianowicie, zdania łączone jakimkolwiek spójnikiem nie muszą być ze sobą powiązane treściowo. W podanych przykładach łączone zdania nie były niezależne. Na przykład (7) i (8) wyrażają pewną własność liczby 10. Jednakże równie dobrze możemy połączyć jakimkolwiek z podanych spójników dwa zdania nie mające ze sobą żadnego związku treściowego, jak np.

(11) 10 jest liczbą parzystą *i* 7 jest liczbą pierwszą,

(12) 10 jest liczbą parzystą *lub* 7 jest liczbą pierwszą,

(13) 10 jest liczbą parzystą *wtedy i tylko wtedy, gdy* 7 jest liczbą pierwszą,

(14) *Jeżeli* 10 jest liczbą parzystą, *to* 7 jest liczbą pierwszą.

O ile składnia zdania (11) nie razi naszego poczucia poprawności, to zdanie (12) z punktu widzenia gramatycznego wydaje się nieco dziwne. W języku potocznym używamy bowiem spójnika „lub", gdy oba zdania składowe alternatywy mają jednakowy podmiot, bądź jednakowe orzeczenie, jak np.:

10 jest liczbą parzystą *lub* 10 jest liczbą pierwszą,

10 jest liczbą parzystą *lub* 7 jest liczbą parzystą.

Podobne uwagi można uczynić odnośnie zdań (13) i (14).

A więc w logice za poprawne uważa się nawet zdanie

10 jest liczbą parzystą *lub* 13-go lutego 1965 r. jest sobota,

czy też zdanie

10 jest liczbą parzystą *wtedy i tylko wtedy, gdy* 13 lutego 1965 r. jest sobota,

bądź zdanie

jeżeli 10 jest liczbą parzystą, *to* 13 lutego 1965 r. jest sobota,

chociaż w języku potocznym tego rodzaju konstrukcje zdań nie są używane. Również w logice nie ma na ogół potrzeby rozważania zdań złożonych, których człony nie wiążą się ze sobą. Ponadto, ponieważ na ogół stosujemy rachunek zdań do teorii matematycznych, więc mało mamy przypadków łączenia spójnikami zdań bardzo się różniących treściowo, tak jak w podanych przykładach. Jednakże dla uniknięcia trudności związanych z dokładnym określeniem kiedy dwa zdania możemy łączyć spójnikiem, przyjmujemy, że spójnikami można łączyć jakiekolwiek zdania niezależne jak i zależne, otrzymując w ten sposób również zdanie zbudowane poprawnie(1).

Streszczenie

W zdaniach teorii matematycznych zasadniczą rolę grają spójniki międzyzdaniowe:

... i ...
... lub ...
... wtedy i tylko wtedy, gdy ...
jeżeli ..., to ...
nie ...

zwane też funktorami zdaniotwórczymi. Za pomocą tych spójników możemy z jakichkolwiek zdań, niekoniecznie związanych ze sobą treściowo, tworzyć nowe zdania poprawne.

(1) Omawiane tu sprawy są wyczerpująco wyjaśnione w podręczniku A. Grzegorczyka [1955], str. 65—75 oraz A. Mostowskiego, str. 7—13.

§ 7. ZDANIA I SCHEMATY ZDAŃ

W podanych w poprzednim paragrafie przykładach zdań złożonych spójnik występował tylko jeden raz. Nic nie stoi oczywiście na przeszkodzie, aby zdania złożone łączyć dalej spójnikami, otrzymując w rezultacie zdania złożone z więcej niż dwu zdań prostych. Aby badanie struktury takich zdań uczynić bardziej przejrzystym, nie będziemy posługiwali się przykładami zdań, a wprowadzimy pewne skróty. Zdania będziemy oznaczali małymi literami łacińskimi, np. p, q, r. Litery te będziemy nazywali *zmiennymi zdaniowymi*. Będą one grały podobną rolę jak zmienne liczbowe. Wprowadzimy również specjalne oznaczenia symboliczne na spójniki zdaniowe, według tabelki poniżej(1):

i	$\&$
lub	$\vee$
i wtedy i tylko wtedy, gdy	$\equiv$
jeżeli..., to	$\Rightarrow$
nie	$\sim$

W przyjętej symbolice omawiane typy zdań w poprzednim paragrafie zapiszemy:

alternatywa	$p \vee q$,
koniunkcja	$p \,\&\, q$,
równoważność	$p \equiv q$,
implikacja	$p \Rightarrow q$,
negacja	$\sim p$.

Oczywiście dwa dowolne zdania złożone możemy ponownie łączyć jednym ze spójników, otrzymując bardziej złożone zdania itd. Na przykład:

$$(p \vee q)\,\&\,r,$$

$$(\sim p\,\&\,r) \equiv \sim q,$$

$$(p \equiv q)\,\&\,(p \Rightarrow q).$$

(1) W literaturze spotykane są również inne sposoby oznaczania spójników logicznych

Wyrażenia jak wyżej, w których zamiast zdań występują zmienne zdaniowe, nie są oczywiście zdaniami, chociażby z tego powodu, że w każdym zdaniu musimy umieć wyróżnić podmiot i orzeczenie. W powyższych zaś wyrażeniach uczynić tego nie możemy. Jeżeli natomiast litery zastąpimy jakimiś zdaniami (prostymi lub złożonymi), to z wyrażeń tych otrzymamy już zdania. Wyrażenia te grają więc rolę jak gdyby schematów zdań, z których możemy otrzymać zdania o określonej strukturze. W schematach tych szczególnie uwypuklona jest rola spójników zdaniowych, tak że od razu widać w jaki sposób poszczególne spójniki łączą ze sobą zdania elementarne. Schematy zdań przypominają formuły arytmetyczne, takie jak np.

$$(x+y)-z,$$

$$(x\cdot y)+z,$$

i z tego powodu są nazywane *formułami rachunku* zdań lub krótko *formułami*. Pojęcie formuły możemy również określić formalnie przez podanie definicji indukcyjnej:

1. Litery p, q, r, ... są formułami.
2. Jeżeli Φ i Ψ są dowolnymi formułami, to również formułami są następujące wyrażenia:

$$(\Phi)\,\&\,(\Psi),$$

$$(\Phi) \vee (\Psi),$$

$$(\Phi) \equiv (\Psi),$$

$$(\Phi) \Rightarrow (\Psi).$$

3. Jeżeli Φ jest formułą, to $\sim(\Phi)$ jest również formułą.

Na przykład, jeżeli zgodnie z punktem 1 definicji, litery p i q są formułami, to na podstawie punktu 2 formułą jest również wyrażenie $(p) \vee (q)$, zgodnie zaś z punktem 3 formułą jest $\sim\big((p) \vee (q)\big)$, dalej zaś na podstawie punktów 1 i 2 formułą będzie również wyrażenie $\Big(\sim\big((p) \vee (q)\big)\Big) \equiv (p)$.

W formułach zbudowanych według podanej definicji występują wszystkie nawiasy, co przy dłuższych formułach utrudnia nieco ich czytanie. Przyjmiemy więc, jak to się czyni w algebrze czy w arytmetyce, że wszystkich nawiasów nie piszemy w formułach, a tylko te które są niezbędne

dla jednoznacznego ich odczytania. Zamiast więc np. pisać $\big(\sim\big((p) \vee (q)\big)\big) \equiv (p)$, napiszemy $\sim(p \vee q) \equiv p$.

Ogólnie przyjmiemy następujące zasady opuszczania nawiasów:

1. Pojedynczej zmiennej nie piszemy w nawiasach.
2. Negując wyrażenie, które jest już poprzedzone symbolem negacji nie ujmuje się tego wyrażenia w nawiasy.
3. Przyjmiemy, że spójniki wiążą w następującej kolejności

$$\sim,\ \&,\ \Rightarrow,\ \equiv,\ \vee.$$

Powyższe zasady zilustrujemy przykładami:

$$p \,\&\, q \vee r \quad \text{oznacza} \quad \big((p \,\&\, q) \vee r\big),$$

gdyż iloczyn wiąże mocniej niż suma.

$$p \,\&\, q \equiv r \quad \text{oznacza} \quad \big((p \,\&\, q) \equiv r\big),$$

gdyż równoważność wiąże słabiej niż iloczyn.

$$p \Rightarrow q \equiv r \quad \text{oznacza} \quad \big((p \Rightarrow q) \equiv r\big),$$

gdyż implikacja wiąże mocniej niż równoważność.

Streszczenie

Schematy zdań albo inaczej formuły rachunku zdań, są to wyrażenia zbudowane ze zmiennych zdaniowych oraz funktorów zdaniotwórczych, tzn. spójników logicznych, podobnie jak formuły algebraiczne — zbudowane ze zmiennych liczbowych oraz symboli operacji arytmetycznych.

Zadania

1. Określmy pojęcie długości formuły następująco. Zmienne zdaniowe mają długość 1; Jeżeli formuły Φ i Ψ mają długości m i n, to formuła $\Phi \circ \Psi$, gdzie $\circ$ jest spójnikiem zdaniowym dwuargumentowym, ma długość $m+n+1$, formuła $\sim \Phi$ zaś ma długość $m+1$.

a) Obliczyć długość formuł:

$$\sim p \vee (q \equiv r) \Rightarrow \sim(\sim p),$$

$$[(p \Rightarrow q) \Rightarrow (p \Rightarrow r)] \Rightarrow (p \Rightarrow r).$$

b) Udowodnić przez indukcję, że długość formuły jest równa ilości znaków pomniejszonej o ilość nawiasów.

c) Udowodnić, że dla skończonej ilości zmiennych zdaniowych (równej $S \geqslant 5$) liczba formuł o długości n zapisanych za pomocą spójników $\&, \vee, \equiv, \Rightarrow, \sim$ jest $< S^n$.

2. a) Usunąć zbędne nawiasy z formuł:

$$[p \,\&\, (p \equiv q)] \vee (\sim p),$$

$$p \Rightarrow \big(q \Rightarrow (r \Rightarrow (s \Rightarrow t))\big),$$

$$\sim([(p \Rightarrow q) \equiv r] \vee p).$$

b) Ile różnych formuł można otrzymać z formuł z punktu a), zmieniając ustawienie nawiasów i ich ilość?

c) Spójnik zdaniowy $\circ$ występujący w formule $f(p, q, \ldots, t)$ nazywamy *spójnikiem głównym*, jeżeli:

albo: istnieją formuły $g(p, q, \ldots, t)$ i $h(p, q, \ldots, t)$ takie, że formuła

$$g(p, q, \ldots, t) \circ h(p, q, \ldots, t)$$

jest formułą $f(p, q, \ldots, t)$,

albo: $\circ$ jest negacją $\sim$ i istnieje formuła $g(p, q, \ldots, t)$ taka, że formuła $\sim g(p, q, \ldots, t)$ jest formułą $f(p, q, \ldots, t)$.

Wskazać spójnik główny w formułach z punktu a).

3. a) Pisząc: $\vee pq$ zamiast $p \vee q$,

$\& pq$ zamiast $p \,\&\, q$,

$\Rightarrow pq$ zamiast $p \Rightarrow q$,

$\equiv pq$ zamiast $p \equiv q$

i zwyczajnie $\sim p$,

zapisać formuły

$$\sim(p \Rightarrow (\sim q) \equiv r),$$

$$p \Rightarrow p \vee q.$$

b) Udowodnić, że w każdej przepisanej w taki sposób formule spójnik główny będzie stał na pierwszym miejscu.

§ 8. PRAWDZIWOŚĆ ZDAŃ ZŁOŻONYCH

W poprzednich paragrafach zajmowaliśmy się budową zdań występujących w teoriach matematycznych. Podane struktury zdań nie wyczerpują wszystkich typów zdań spotykanych w matematyce (z dalszymi rodzajami zdań zapoznamy się w następnym rozdziale), jednakże stanowią one podstawę wszelkich wypowiedzi matematycznych. Jak już to stwierdziliśmy, głównym celem logiki jest badanie kryteriów prawdziwości zdań w zależności od ich struktury. Chodzi bowiem o to, abyśmy o prawdziwości jakiegoś zdania interesującej nas teorii mogli sądzić tylko na podstawie jego struktury nie wchodząc w jego sens i znaczenie. Fragment logiki zwany rachunkiem zdań podaje nam metody badania prawdziwości zdań, które składają się ze zdań składowych połączonych odpowiednio, podanymi w poprzednim paragrafie spójnikami zdaniowymi. Punktem wyjściowym do tych rozważań jest określenie jak zależy prawdziwość zdania złożonego z dwóch zdań, od prawdziwości zdań składowych oraz użytego do ich połączenia spójnika zdaniowego. (Oczywiście w przypadku zastosowania spójnika „nie" do budowy nowego zdania ze zdania danego, prawdziwość negacji zależy tylko od jednego zdania składowego). Zagadnienie to omówimy kolejno dla wszystkich podanych spójników.

1. *Alternatywa*, $p \vee q$, jest prawdziwa wtedy i tylko wtedy, gdy choć jedno ze zdań p, q jest prawdziwe. Na przykład zdanie

10 jest liczbą nieparzystą *lub* 10 jest liczbą parzystą,

jest prawdziwe, gdyż prawdziwe jest zdanie

10 jest liczbą parzystą.

Natomiast zdanie

10 jest liczbą pierwszą *lub* 10 jest liczbą większą od 20

jest fałszywe, gdyż oba zdania wchodzące w skład alternatywy są fałszywe. Oczywiście, jeżeli oba zdania p, q alternatywy są prawdziwe, to alternatywa jest również prawdziwa. Dla krótkości zapisu wygodnie jest używać następujących skrótów: Zamiast „zdanie p jest prawdziwe", zapiszemy $p = 1$, oraz zamiast „zdanie p jest fałszywe", zapiszemy $p = 0$. Symbole 0 i 1 grają w naszym zapisie rolę słów „fałszywe" i „prawdziwe". Są one też

nazywane *wartościami logicznymi*. Tak więc każda zmienna zdaniowa może przyjmować jedną z dwu wartości logicznych „prawda" i „fałsz" lub w skrócie 1 i 0. Prawdziwość alternatywy możemy teraz scharakteryzować następująco:

Alternatywa $p \vee q = 1$, wtedy i tylko wtedy, gdy $p = 1$ lub[1] $q = 1$.

Własność tę wygodnie jest ująć w postaci tabelki:

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

W języku potocznym, gdy dwa zdania są niezależne treściowo i oba są prawdziwe, jak już zauważyliśmy, zazwyczaj nie łączymy ich spójnikiem „lub". Na przykład konstrukcja zdania

Janka jest ładna lub Ina ma zielone oczy

jakkolwiek poprawna z gramatycznego punktu widzenia jest jednak nie używana, nie ma bowiem potrzeby stosowania takiej konstrukcji. Fakty wyrażone w tym zdaniu powiedzielibyśmy w postaci dwu oddzielnych zdań. W matematyce natomiast zdarza się często taka sytuacja, że znamy prawdziwość alternatywy, nie wiemy natomiast jeszcze, które ze zdań składowych jest prawdziwe. Ponieważ nie możemy nic powiedzieć w takim przypadku o prawdziwości zdań p i q, musimy więc w rozumowaniu posługiwać się ich alternatywą.

2. *Koniunkcja*, $p \& q$, jest prawdziwa wtedy i tylko wtedy, gdy oba zdania składowe p oraz q są prawdziwe. Tabelka dla koniunkcji ma postać:

p	q	$p \& q$
0	0	0
0	1	0
1	0	0
1	1	1

[1] Należy zwrócić uwagę, że słowo „lub" jest tu słowem z języka potocznego.

Nie wymaga ona komentarza, gdyż w języku potocznym posługujemy się spójnikiem „i" w sposób identyczny, tzn. łączymy nim wtedy dwa zdania, gdy chcemy wyrazić przekonanie, że oba zdania koniunkcji są prawdziwe.

3. *Implikacja* jest spójnikiem, którego zrozumienie sprawia największe kłopoty, dlatego radzimy zwrócić na nią szczególną uwagę. Nie będziemy starali się podać dlaczego przyjęto taką a nie inną definicję prawdziwości implikacji w logice, gdyż sprawy te są doskonale przedstawione w wielu innych podręcznikach logiki (patrz A. Grzegorczyk [1955], A. Mostowski), a nie sądzimy, aby udało się nam z równą ścisłością uzasadnić stanowisko jakie zajmują w tej sprawie logicy. Podkreślamy tylko, że używanie zwrotu „jeżeli ..., to" w zdaniach teorii matematycznych odbiega od posługiwania się nim w języku potocznym, co jest źródłem wielu nieporozumień. Wynikają one stąd, że w języku potocznym łączenie dwu zdań zwrotem „jeżeli ..., to" jest możliwe jedynie w bardzo szczególnych okolicznościach, natomiast w zdaniach języków teorii matematycznych, na zdania łączone tym spójnikiem nie nakładamy żadnych warunków, tzn. możemy nim łączyć jakiekolwiek dwa zdania, jak np.

(1) *Jeżeli* 10 jest liczbą pierwszą, *to* 2 jest większe od 5.

Implikację, $p \Rightarrow q$, uważamy za fałszywą, jedynie w tym przypadku, jeżeli zdanie p jest prawdziwe a zdanie q — fałszywe. W każdym innym przypadku implikację uważamy za prawdziwą.

Powyższą własność możemy wyrazić w postaci tabelki:

p	q	$p \Rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

Zgodnie z podaną definicją prawdziwości implikacji, zdanie (1) logicy uważają za zdanie prawdziwe.

Zdanie p nazywamy *poprzednikiem* albo *założeniem implikacji*, natomiast zdanie q — *następnikiem* albo *tezą*. Bardzo ważna jest z matematycznego punktu widzenia następująca własność implikacji: *jeżeli wiemy, że implikacja jest prawdziwa oraz prawdziwe jest założenie (poprzednik), to prawdziwa jest również teza (następnik)*. W rozumowaniach matematycznych bowiem bardzo często zachodzi potrzeba postępowania odwrotnego niż to, które opisujemy obecnie, tzn. nie chodzi nam o znalezienie prawdziwości zdania złożonego w zależności od zdań składowych a odwrotnie: określenie prawdziwości jednego ze zdań składowych, gdy znamy prawdziwość zdania złożonego i prawdziwość drugiego zdania składowego. Implikacja pozwala na wnioskowanie o prawdziwości jej następnika na podstawie prawdziwości całego zdania oraz poprzednika. Fakt ten jest wykorzystany w tzw. regule odrywania, o której będzie mowa w § 14.

4. *Równoważność*, $p \equiv q$, jest prawdziwa wtedy i tylko wtedy, gdy oba zdania p i q są albo jednocześnie prawdziwe, albo jednocześnie fałszywe.

Tabelka równoważności będzie więc miała postać:

p	q	$p \equiv q$
0	0	1
0	1	0
1	0	0
1	1	1

Sens tego spójnika jest oczywisty.

5. *Negacja*, $\sim p$, jest prawdziwa, jeżeli zdanie p jest fałszywe i odwrotnie. Tabelka negacji jest więc bardzo prosta

p	$\sim p$
0	1
1	0

Rozumienie jej nie przedstawia trudności.

6. Obecnie możemy już określić wartość logiczną dowolnego zdania złożonego. Wystarczy do odpowiadającej mu formuły logicznej wstawić wartości zdań elementarnych i posługując się tabelkami charakteryzującymi spójniki logiczne, obliczyć wartość logiczną zdania badanego, tj. sprawdzić czy jest ono prawdziwe, czy fałszywe. Postępowanie przy obli-

czaniu wartości logicznej zdania złożonego jest podobne do wykonywania zwykłego rachunku liczbowego, z tą różnicą, że w obecnym przypadku rolę działań arytmetycznych grają spójniki zdaniowe, których działanie na wartościach 0 i 1 określone jest tabelkami. Na przykład jeżeli $p = 1$, $q = 1$ oraz $r = 0$, to

$$\sim(p \,\&\, q) \equiv r = \sim(1 \,\&\, 1) \equiv 0 = 1.$$

Streszczenie

Jeżeli w formule rachunku zdań na miejsce zmiennych zdaniowych podstawimy zdania, to w rezultacie otrzymamy zdanie złożone. Jeżeli natomiast na miejsce zmiennych zdaniowych w formule podstawimy wartości logiczne interesujących nas zdań, to wykonując obliczenie zgodnie z tabliczkami spójników logicznych, otrzymamy wartość logiczną zdania złożonego. Badanie prawdziwości zdań sprowadza się więc do wykonywania prostych rachunków, bez potrzeby wnikania w sens badanego zdania.

Zadania

1. Dla jakich wartości logicznych p, q, r, niżej podane formuły przyjmują wartość 0:

$$(p \Rightarrow q) \Rightarrow r,$$

$$(p \Rightarrow \sim p) \Rightarrow p,$$

$$(p \Rightarrow \sim p) \Rightarrow \sim p,$$

$$p \vee q \equiv r,$$

$$p \,\&\, q \equiv q.$$

2. a) Uzasadnić, że jeżeli zdanie ma postać implikacji,

$$f(p, q, r) \Rightarrow g(p, q, r),$$

to by sprawdzić jego prawdziwość wystarcza sprawdzić tylko te podstawienia 0 i 1 na p, q, r, przy których jednocześnie poprzednik $f(p, q, r)$ jest prawdziwy, następnik zaś fałszywy.

b) Sprawdzić, czy zdania

$$p \Rightarrow p,$$

$$(p \Rightarrow \sim p) \Rightarrow q,$$

$$((p \Rightarrow q) \Rightarrow (q \Rightarrow r)) \Rightarrow (p \Rightarrow r),$$

$$p \Rightarrow (p \Rightarrow q)$$

są prawdziwe dla dowolnych wartości logicznych p, q oraz r.

§ 9. FORMUŁY ZAWSZE PRAWDZIWE. TAUTOLOGIE

Podstawiając w dowolnej formule za zmienne ich wartości logiczne i wykonując występujące w formule operacje logiczne, otrzymujemy wartość logiczną całego zdania złożonego(1).

Przy podanej metodzie obliczania wartości logicznej zdań złożonych, istnieją w rachunku zdań takie formuły — jak to łatwo zauważyć — których wartość logiczna nie zależy od wartości logicznych występujących w nich zmiennych i zawsze jest równa 0 albo też 1. Na przykład wartość logiczna formuły

$$p \vee \sim p = 1,$$

jeżeli bowiem $p = 0$, to

$$0 \vee \sim 0 = 0 \vee 1 = 1$$

oraz jeżeli $p = 1$, to

$$1 \vee \sim 1 = 1 \vee 0 = 1.$$

Natomiast wartość logiczna formuły

$$p \,\&\, \sim p = 0.$$

W logice szczególną rolę odgrywają formuły, których wartość logiczna jest stale równa 1, niezależnie od wartości logicznych występujących w nich zmiennych zdaniowych. Formuły takie są nazywane *tautologiami logicz-*

(1) Z uwagi na podobieństwo roli spójników $\sim$, $\vee$, &, w obliczaniu wartości logicznej zdania czy formuły, do roli operacji arytmetycznych „+", „—" w wykonywaniu rachunków liczbowych, spójniki zdaniowe, w takiej sytuacji, można nazwać *operacjam logicznymi.*

nymi albo *prawami logicznymi.* Jeśli jakaś formuła jest tautologią, będziemy pisali przed nią symbol „ $\vdash$ ", np.

$$\vdash p \vee \sim p.$$

Oczywiście jeżeli wartość jakiejś formuły jest zawsze równa zeru, to jej negacja będzie tautologią logiczną, np.

$$\vdash \sim (p \,\&\, \sim p).$$

Można sprawdzić, że następujące formuły są tautologiami logicznymi:

(1) $p \Rightarrow (q \Rightarrow p)$ *prawo symplifikacji,*

(2) $(p \Rightarrow \sim p) \Rightarrow \sim p$ } *prawa Claviusa,*

(3) $(\sim p \Rightarrow p) \Rightarrow p$ }

(4) $(p \Rightarrow q) \Rightarrow (\sim q \Rightarrow \sim p)$ *prawo transformacji,*

(5) $(p \,\&\, \sim p) \Rightarrow q$ *prawo Dunsa Scotusa,*

(6) $(p \Rightarrow q) \Rightarrow \big((q \Rightarrow r) \Rightarrow (p \Rightarrow r)\big)$ *prawo sylogizmu.*

Tautologie stanowią schematy poprawnych rozumowań, z których korzystamy przeprowadzając jakiekolwiek wnioskowanie, niekoniecznie na terenie matematyki, chociaż nie zdajemy sobie na ogół sprawy z tego faktu.

Na przykład bardzo często w rozumowaniach korzystamy z tautologii

(7) $$\vdash (p \vee \sim p)$$

zwanej *prawem wyłączonego środka.* Prawo to stwierdza, że ze zdań p oraz $\sim p$ co najmniej jedno jest prawdziwe. Jeżeli więc stwierdzamy, że „dziś jest poniedziałek lub nieprawda, że dziś jest poniedziałek", to co najmniej jedno ze zdań składowych tej alternatywy musi być prawdziwe. Inna możliwość nie istnieje. Jest tak jak głosi pierwszy składnik alternatywy lub tak jak głosi drugi składnik.

Tautologia

(8) $$\vdash \sim (\sim p \,\&\, p)$$

jest nazywana *prawem wyłączonej sprzeczności.* Stwierdza ono, że ze zdań p oraz $\sim p$ co najwyżej jedno jest prawdziwe. Prawo wyłączonego środka

oraz prawo wyłączonej sprzeczności stwierdzają więc, że z dwu zdań sprzecznych p oraz $\sim p$, dokładnie jedno jest zdaniem prawdziwym.

Aby wyjaśnić znaczenie tautologii rozpatrzmy jeszcze następujący przykład. Zdanie

(9) jeżeli liczba a jest większa od zera i liczba b jest większa od zera, to liczba $a \cdot b$ jest większa od zera

jest prawdziwe. Ma ono schemat

(10) $$p \,\&\, q \Rightarrow r,$$

gdzie p oznacza zdanie

(11) liczba a jest większa od zera,

q oznacza zdanie

(12) liczba b jest większa od zera,

r zaś

(13) liczba $a \cdot b$ jest większa od zera.

Formuła (10) nie jest tautologią. Gdy położymy $p = 1$, $q = 1$, zaś $r = 0$, wtedy formuła (10) przyjmuje wartość 0. O prawdziwości zdania (9) wnioskujemy nie na podstawie jego kształtu, a na podstawie sensu matematycznego zdań (11), (12) i (13). Zdanie (9) jest twierdzeniem matematyki, ale nie jest szczególnym przypadkiem tautologii.

Inaczej sprawa przedstawia się ze zdaniem

(liczba a jest większa od zera)

lub

nie (liczba a jest większa od zera).

Zdanie to wyraża twierdzenie matematyki, które jest szczególnym przypadkiem prawa wyłączonego środka.

Streszczenie

Formuły rachunku zdań, zwane tautologiami logicznymi, albo prawami logicznymi, są to formuły rachunku zdań, które niezależnie od wartości występujących w nich zmiennych mają zawsze wartość logiczną 1. Sprawdzenie czy jakaś formuła jest tautologią jest więc bardzo proste i polega

na wykonaniu wszystkich możliwych podstawień wartości logicznych za zmienne zdaniowe i obliczeniu dla każdego podstawienia wartości logicznej całej formuły. Jeżeli dla każdego podstawienia wartość logiczna formuły jest równa 1, to formuła ta jest tautologią(1).

Zadania

1. a) Udowodnić, że formuły (1)-(6) ze strony 32 są tautologiami logicznymi.

2. a) Udowodnić tautologie:

$$((p \equiv q) \equiv r) \equiv (p \equiv (q \equiv r)),$$
$$(p \equiv q) \equiv (q \equiv p).$$

b) Sprawdzić następujące tautologie

$$\vdash (p \equiv q) \Rightarrow (p \Rightarrow q),$$
$$\vdash (p \Rightarrow q) \Rightarrow ((q \Rightarrow p) \Rightarrow (p \equiv q)),$$
$$\vdash (p \Rightarrow q) \,\&\, (\sim q \Rightarrow \sim p) \equiv (p \equiv q),$$
$$\vdash (p \Rightarrow q) \equiv \sim p \vee q,$$
$$\vdash (p \Rightarrow q) \equiv \sim (p \,\&\, \sim q).$$

c) Udowodnić, że formuła $f(p, q, r, \ldots)$ zbudowana tylko ze znaków „$\equiv$" równoważności, zawierająca każdą zmienną parzystą ilość razy jest tautologią.

d) Opierając się na tautologiach udowodnionych w punkcie a), udowodnić równoważność

$$[f(p_1, q_1, \ldots, r_1) \equiv g(p_2, q_2, \ldots, r_2)] \equiv [(\ldots (p_1 \equiv p_1) \equiv \ldots \equiv (r_1 \equiv r_1)) \equiv$$
$$\equiv (\ldots (p_2 \equiv p_2) \equiv \ldots \equiv (r_2 \equiv r_2))],$$

gdzie $f(p, q, r, \ldots)$ oraz $g(q, r, \ldots)$ są formułami zbudowanymi ze znaków równoważności.

e) Sprowadzić tautologię

$$\big(((p \equiv r) \equiv s) \equiv p\big) \equiv (r \equiv s).$$

do postaci takiej jak podana w punkcie d).

f) Udowodnić, że jeżeli wyrażenie $f(p, g, r, \ldots)$ zbudowane tylko za pomocą spójników równoważności jest tautologią, to każda zmienna występuje w nim parzystą ilość razy.

---

(1) Takie sprawdzanie jest często dość kłopotliwe, gdyż wymaga stosunkowo dużej liczby operacji dla zbadania czy dana formuła jest tautologią. Często stosuje się uproszczone metody, pozwalające na szybsze sprawdzenie, czy formuła jest tautologią aniżeli za pomocą wszystkich możliwych podstawień. Por. zad. 2 § 7.

§ 10. PRZEKSZTAŁCANIE FORMUŁ RACHUNKU ZDAŃ

W paragrafie tym podamy kilka praw logicznych potrzebnych do przekształcania formuł rachunku zdań oraz przykłady ich zastosowania w upraszczaniu formuł. Dla wygody w tym paragrafie, a także w paragrafach 11-13, za formuły logiczne będziemy uważali także dwa wyrażenia: 0, które czytamy „fałsz" oraz 1, które czytamy „prawda".

1. *Prawo podwójnego przeczenia*:

$$\vdash \sim \sim p \equiv p.$$

2. *Prawa tautologii*:

$$\vdash (p \vee p) \equiv p,$$
$$\vdash (p \,\&\, p) \equiv p.$$

3. *Prawa przemienności*:

$$\vdash (p \vee q) \equiv (q \vee p),$$
$$\vdash (p \,\&\, q) \equiv (q \,\&\, p).$$

4. *Prawa łączności*:

$$\vdash \big(p \vee (q \vee r)\big) \equiv \big((p \vee q) \vee r\big),$$
$$\vdash \big(p \,\&\, (q \,\&\, r)\big) \equiv \big((p \,\&\, q) \,\&\, r\big).$$

5. *Prawo rozdzielności mnożenia względem dodawania*:

$$\vdash p \,\&\, (q \vee r) \equiv (p \,\&\, q) \vee (p \,\&\, r).$$

6. *Prawo rozdzielności dodawania względem mnożenia*:

$$\vdash p \vee (q \,\&\, r) \equiv (p \vee q) \,\&\, (p \vee r).$$

7. *Prawa de Morgana*:

$$\vdash \sim (p \vee q) \equiv \sim p \,\&\, \sim q,$$
$$\vdash \sim (p \,\&\, q) \equiv \sim p \vee \sim q.$$

Sens wszystkich tych praw jest oczywisty. Warto jedynie zwrócić uwagę, że prawa 3, 4 i 5 przypominają znane nam prawa z arytmetyki, natomiast odpowiednika prawa 6 w arytmetyce nie ma, nie możemy bowiem napisać:

$$x + (y \cdot z) = (x + y) \cdot (x + z).$$

Również prawa 2 nie mają odpowiednika w arytmetyce, gdyż

$$x+x=2x,$$
$$x\cdot x=x^2.$$

Prawo 1 zaś przypomina następujące prawo znane z arytmetyki:

$$-\big(-(x)\big)=x.$$

Natomiast pozostałe prawa nie mają odpowiedników w arytmetyce.

Dla przykładu uprośćmy następującą formułę logiczną

(1) $$\big((p\vee q)\&\sim p\big)\&q.$$

Stosując do pierwszej z lewej strony koniunkcji prawo 5, otrzymamy

(2) $$\big((p\&\sim p)\vee(q\&\sim p)\big)\&q.$$

Ponieważ

$$(p\&\sim p)=0$$

na podstawie prawa wyłączonej sprzeczności, więc formułę (2) możemy napisać w postaci:

(3) $$\big(0\vee(q\&\sim p)\big)\&q.$$

Z definicji alternatywy: formułę (3) możemy zastąpić równoważną jej formułą

(4) $$(q\&\sim p)\&q;$$

stąd z praw 3, 4 i 2 przekształcamy formułę (4) na formułę

(5) $$\sim p\&q;$$

ostatecznie możemy więc napisać

(6) $$\big((p\vee q)\&\sim p\big)\&q=\sim p\&q.$$

Znak równości należy tu rozumieć tak, że wartość logiczna formuły stojącej po lewej stronie jest równa wartości logicznej formuły stojącej po prawej stronie. Z podanych przez nas rozważań wynika, że tak jest w istocie. Więc formuła

(7) $$\big((p\vee q)\&\sim p\big)\&q\equiv\sim p\&q$$

jest tautologią logiczną.

Streszczenie

Prawa logiki pozwalają na przekształcanie formuł rachunku zdań, podobnie jak prawa arytmetyki pozwalają na przekształcenie formuł arytmetycznych. Niektóre prawa logiki przypominają prawa znane z arytmetyki, jak np. prawo łączności, jednakże niektóre prawa logiki nie mają odpowiedników w arytmetyce.

Zadania

1. Przekształcić formułę

$$q\Rightarrow(p\&\sim p)$$

na formułę $\sim q$.

2. Przekształcić formuły

$$p\Rightarrow q,\quad \sim(p\Rightarrow q),\quad p\Rightarrow(p\Rightarrow q)$$

na formuły zawierające

a) alternatywę i negację; b) koniunkcję i negację.

3. Przekształcić formuły na formuły zawierające implikacje i negacje:

$$p\equiv q,\quad p\vee q,\quad p\&q,\quad \sim p\vee q,\quad p\&\sim q.$$

4. Uprościć formuły

a) $$(p\&q)\vee\sim(p\vee q);$$

b) $$(p\&\sim q)\vee\sim(p\vee q)$$

na formuły o najmniejszej długości (por. zad. 1 z § 7), zawierające jakiekolwiek ze spójników $\vee, \&, \equiv, \Rightarrow, \sim$.

5. a) Sprawdzić tautologie (1)-(7) z tego paragrafu.

b) Z tautologii (2)-(4) wywnioskować, że każdą formułę $f(p,q,r,\ldots,s)$ zbudowaną ze zmiennych $p,q,r,\ldots,s$ oraz symbolu alternatywy „$\vee$" można przekształcić na równoważną jej formułę

(*) $$p\vee q\vee r\vee\ldots\vee s,$$

gdzie ustawienie nawiasów jest obojętne. Podobnie dla funktora koniunkcji „&".

c) Udowodnić, że formuła

(**) $$g\vee h\vee k\vee\ldots\vee m$$

zbudowana ze zmiennych $p, q, r, \ldots$ tak, że każda z liter występujących w (**) jest albo zmienną, albo jej negacją, jest tautologią wtedy i tylko wtedy, gdy jakaś zmienna występuje co najmniej raz zanegowana i co najmniej raz niezanegowana.

d) Pokazać, że negacja formuły (**) nie może być tautologią.

§ 11. INNE SPÓJNIKI

Podane przykłady spójników nie wyczerpują wszystkich możliwych spójników, które można zastosować do łączenia dwóch zdań. Na przykład możemy wprowadzić spójnik „$\not\equiv$", zwany *różnicą symetryczną* zdań p i q, który ma następującą tabelkę:

p	q	$p \not\equiv q$
0	0	0
0	1	1
1	0	1
1	1	0

Różnica symetryczna zdań jest więc prawdziwa wtedy i tylko wtedy, gdy oba zdania p i q mają różne wartości logiczne. Zdanie $p \not\equiv q$ czytamy zazwyczaj „albo p, albo q".

Innym ciekawym przykładem spójnika jest *jednoczesne zaprzeczenie* zdań p i q. Spójnik ten będziemy oznaczać symbolem „/" i zdanie p/q będziemy czytali „ani p, ani q". Tabliczka tego spójnika jest następująca:

p	q	p/q
0	0	1
0	1	0
1	0	0
1	1	0

Ponieważ spójniki interesują nas z punktu widzenia wartości logicznej zdania złożonego za ich pomocą — w zależności od zdań składowych można więc napisać 2^4 różnych tabliczek charakteryzujących spójniki.

Możliwych jest więc 16 różnych spójników zdaniowych dwuargumentowych. Wszystkie te spójniki są podane w tablicy poniżej:

Nr kolejny	Tabliczka	Oznaczenie
0	0000	0
1	0001	$p \,\&\, q$
2	0010	$p \nearrow q$
3	0011	p
4	0100	$p \nwarrow q$
5	0101	q
6	0110	$p \not\equiv q$
7	0111	$p \vee q$
8	1000	p/q
9	1001	$p \equiv q$
10	1010	$\sim q$
11	1011	$q \Rightarrow p$
12	1100	$\sim p$
13	1101	$p \Rightarrow q$
14	1110	$p \,\backslash\, q$
15	1111	1

Spójników jednoargumentowych jest $2^2 = 4$. Oto one

Nr kolejny	Tabliczka	Oznaczenie
0	00	0
1	01	p
2	10	$\sim p$
3	11	1

W drugiej kolumnie podane są wartości zdania otrzymanego przez zastosowanie funktora do zdania p kolejno dla $p = 0$ i $p = 1$. Funktory nr 0 i nr 3, to funktory tworzące ze zdania p odpowiednio zdanie zawsze fałszywe i zdanie zawsze prawdziwe. Funktor nr 1 to funktor tożsamościowy nie zmieniający zdania p, funktor zaś nr 2 to negacja.

Przejdziemy teraz do omawiania tabelki funktorów dwuargumentowych.

W drugiej kolumnie tablicy podano kolejne wartości zdania złożonego za pomocą odpowiedniego spójnika, dla kolejnych wartości zmiennych p i q: 00, 01, 10, 11. Wiersze tej kolumny są więc ostatnimi kolumnami tabelki odpowiedniego spójnika. Na przykład dla implikacji wiersz ten ma postać 1101, gdyż $0 \Rightarrow 0 = 1$, $0 \Rightarrow 1 = 1$, $1 \Rightarrow 0 = 0$, $1 \Rightarrow 1 = 1$.

Spójnik 0 oraz spójnik 15 z dowolnych zdań tworzą odpowiednie zdanie zawsze prawdziwe oraz zdanie zawsze fałszywe. Właściwie nie są one więc spójnikami, jednakże dla uzyskania jednolitości potraktowaliśmy je jako spójniki.

Spójniki 3 oraz 5 przypisują zdaniom p bądź q te same zdanie, natomiast spójniki 10 oraz 12 — ich negacje. Pozostałe wiersze tablicy wyczerpują wszystkie możliwe pozostałe spójniki dwuargumentowe, tj. tworzące z dwóch zdań p oraz q nowe zdanie.

Spójniki podane w tablicy nie są od siebie niezależne. Między nimi istnieją pewne związki. Na przykład równoważność możemy określić za pomocą sumy, iloczynu i negacji w następujący sposób:

$$p \equiv q = (p \& q) \vee (\sim p \& \sim q).$$

Podobnie możemy określić inne spójniki:

$$p \not\equiv q = (p \& \sim q) \vee (\sim p \& q),$$

$$p \Rightarrow q = \sim p \vee q,$$

$$p/q = \sim p \& \sim q,$$

$$p \setminus q = \sim p \vee \sim q.$$

Spójniki, negacja oraz alternatywa, bądź też negacja i koniunkcja, jak również negacja i implikacja wystarczają do określenia wszystkich pozostałych spójników.

Również spójnik 8, p/q, o którym była mowa na str. 38, a także spójnik 14 (zwany *dyzjunkcją Sheffera*) wystarczają do określenia wszystkich pozostałych spójników. Na przykład

$$\sim p = (p/p),$$

$$p \vee q = \big((p/q)/(p/q)\big).$$

Streszczenie

Przypisując każdej parze wartości logicznych 00, 01, 10, 11 na wszystkie sposoby wartości 0 i 1, otrzymamy 16 możliwych spójników logicznych. Nie wszystkie z nich odgrywają rolę w logice. W niektórych zastosowaniach logiki jednakże wygodnie jest mieć możliwość dobrania takich spójników, które są aktualnie przydatne do naszych celów.

Zadania

1. Udowodnić, że za pomocą spójników zdaniowych 0 i $\Rightarrow$ można określić negację, alternatywę, koniunkcję i równoważność (por. zadanie 1 i zadanie 3 z poprzedniego paragrafu).

2. Wypisać wszystkie spójniki, które można określić za pomocą spójników „$\not\equiv$" i „$\sim$,".

3. a) Określić negację, alternatywę i koniunkcję za pomocą spójnika p/q.
b) Określić negację, alternatywę i koniunkcję za pomocą spójnika $p \setminus q$.
c) Wyrazić spójnik $p \setminus q$ przez spójnik p/q i na odwrót.

4. Wyrazić wszystkie spójniki, które można określić przez:
a) implikację i alternatywę,
b) implikację i koniunkcję.

§ 12. POSTACIE NORMALNE

Formułę postaci

$$p_1^{c_1} \& p_2^{c_2} \& \ldots \& p_n^{c_n},$$

gdzie $c_i = 0$ lub $c_i = 1$ oraz $p_i^0 = \sim p_i$ natomiast $p_i^1 = p_i$, będziemy nazywali *koniunkcją elementarną*. Koniunkcja elementarna jest więc iloczynem logicznym n różnych zmiennych zdaniowych zanegowanych bądź nie. W szczególnym przypadku pojedyncza zmienna zdaniowa jest również koniunkcją elementarną. Dla n zmiennych zdaniowych liczba różnych koniunkcji elementarnych wynosi oczywiście 2^n. Na przykład dla $n = 2$ mamy następujące koniunkcje elementarne

$$p^1 \& q^1 = p \& q,$$

$$p^1 \& q^0 = p \& \sim q,$$

$$p^0 \& q^1 = \sim p \& q,$$
$$p^0 \& q^0 = \sim p \& \sim q,$$

Dla $n = 3$ koniunkcje elementarne będą miały postać

$$p^1 \& q^1 \& r^1 = p \& q \& r,$$
$$p^1 \& q^1 \& r^0 = p \& q \& \sim r,$$
$$p^1 \& q^0 \& r^1 = p \& \sim q \& r,$$
$$p^1 \& q^0 \& r^0 = p \& \sim q \& \sim r,$$
$$p^0 \& q^1 \& r^1 = \sim p \& q \& r,$$
$$p^0 \& q^1 \& r^0 = \sim p \& q \& \sim r,$$
$$p^0 \& q^0 \& r^1 = \sim p \& \sim q \& r,$$
$$p^0 \& q^0 \& r^0 = \sim p \& \sim q \& \sim r.$$

Podobnie określimy alternatywę elementarną. Formułę postaci

$$p_1^{1-c_1} \vee p_2^{1-c_2} \vee \ldots \vee p_n^{1-c_n}$$

będziemy nazywali *alternatywą elementarną*, gdzie c_i, jak poprzednio, jest 0 lub 1 oraz $p_i^0 = \sim p_i$, natomiast $p_i^1 = p_i$.

Alternatywa elementarna jest więc sumą logiczną n różnych zmiennych zdaniowych, z których każda może być zanegowana lub nie. Na przykład

$$p^0 \vee q^1 \vee r^0 = \sim p \vee q \vee \sim r,$$

Oczywiście dla n zmiennych zdaniowych istnieje 2^n różnych alternatyw elementarnych. Dla $n = 2$ alternatywy te będą miały postać:

$$p^1 \vee q^1 = p \vee q,$$
$$p^1 \vee q^0 = p \vee \sim q,$$
$$p^0 \vee q^1 = \sim p \vee q,$$
$$p^0 \vee q^0 = \sim p \vee \sim q.$$

Sumę logiczną dowolnej liczby różnych koniunkcji elementarnych n zmiennych zdaniowych, będziemy nazywali *formułą w dyzjunkcyjnej postaci normalnej*. Na przykład formuła

$$(p \& q) \vee (\sim p \& q)$$

jest w dyzjunkcyjnej postaci normalnej. Dodatkowo umawiamy się, że będziemy uważali formułę 0, czyli formułę $f(p, q, \ldots) = 0$ za formułę w dyzjunkcyjnej pos aci normalnej.

Iloczyn logiczny dowolnej liczby różnych alternatyw elementarnych n zmiennych zdaniowych, będziemy nazywali *formułą w koniunkcyjnej postaci normalnej*. Na przykład

$$(p \vee q) \& (p \vee \sim q) \& (\sim p \vee \sim q)$$

jest formułą w koniunkcyjnej postaci normalnej. Formułę 1, czyli formułę $f(p, q, \ldots) = 1$ uważamy również za formułę w koniunkcyjnej po taci normalnej.

TWIERDZENIE 1. *Dowolną formułę rachunku zdań $\Phi(p_1, p_2, \ldots, p_n)$ można przedstawić w następującej postaci:*

$$\Phi(0, 0, \ldots, 0) \& (p_1^0 \& p_2^0 \& \ldots \& p_n^0) \vee \ldots \vee \vee \Phi(1, 1, \ldots, 1) \& (p_1^1 \& p_2^1 \& \ldots \& p_n^1)$$

zwanej dyzjunkcyjną postacią normalną.

Zauważmy, że

$$\Phi(c_1, c_2, \ldots, c_n) \& (p_1^{c_1} \& p_2^{c_2} \& \ldots \& p_n^{c_n})$$

jest równe 0, jeżeli $\Phi(c_1, c_2, \ldots, c_n) = 0$, równe zaś

$$p_1^{c_1} \& p_2^{c_2} \& \ldots \& p_n^{c_n},$$

jeżeli $\Phi(c_1, c_2, \ldots, c_n) = 1$.

PRZYKŁAD. Przedstawimy w dyzjunkcyjnej postaci normalnej jakąkolwiek formułę $\Phi(p, q, r)$, gdzie p, q, r są zmiennymi zdaniowymi a wartości logiczne formuły Φ są dla odpowiednich wartości p, q oraz r, określone następującą tablicą logiczną:

p	q	r	Φ
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Zgodnie z twierdzeniem 1, aby utworzyć dyzjunkcyjną postać normalną tej formuły, musimy utworzyć wszystkie koniunkcje elementarne trzech zmiennych zdaniowych $p^{c_1}\&q^{c_2}\&r^{c_3}$, mnożąc każdą koniunkcję elementarną przez wartość logiczną formuły dla $p=c_1$, $q=c_2$ oraz $r=c_3$. Ponieważ $0\,\&\,p=0$, więc wszystkie iloczyny elementarne, dla których $\Phi(c_1,c_2,c_3)=0$, odpadną i pozostaną tylko te iloczyny elementarne, dla których $\Phi(c_1,c_2,c_3)=1$. Mówiąc prościej należy z tabelki formuły Φ wybrać te wiersze, w których formuła Φ jest równa 1 i dla nich utworzyć iloczyny elementarne. Suma tych iloczynów będzie poszukiwaną formułą w postaci normalnej. W naszym przykładzie iloczyny te będą następujące:

$$p^0\,\&\,q^0\,\&\,r^0=\sim p\,\&\sim q\,\&\sim r,$$
$$p^1\,\&\,q^0\,\&\,r^0=p\,\&\sim q\,\&\sim r,$$
$$p^1\,\&\,q^0\,\&\,r^1=p\,\&\sim q\,\&\,r,$$
$$p^1\,\&\,q^1\,\&\,r^0=p\,\&\,q\,\&\sim r.$$

A więc poszukiwana dyzjunkcyjna postać normalna dowolnej formuły Φ spełniającej tabelkę, będzie

(1) $(\sim p\,\&\sim q\,\&\sim r)\vee(p\,\&\sim q\,\&\sim r)\vee(p\,\&\sim q\,\&\,r)\vee(p\,\&\,q\,\&\sim r)$

Podana przez nas tabelka opisuje wiele formuł równoważnych sobie. Na przykład formułę (1), a także formułę

(2) $$\sim p\,\&\sim(q\vee r)\vee p\,\&\sim(q\,\&\,r),$$

czy też formułę

(3) $$\sim(q\vee r)\vee p\,\&(q\not\equiv r),$$

względnie formułę (4) podaną przy końcu paragrafu.

Wybranie postaci normalnej wyróżnia wśród wszystkich formuł równoważnych jednego reprezentanta o stosunkowo prostej postaci.

TWIERDZENIE 2. *Dowolną formułę rachunku zdań można przedstawić w postaci*

$$\Phi(0,0,\ldots,0)\vee(p_1^1\vee p_2^1\vee\ldots\vee p_n^1)\,\&\ldots\,\&\,\&\,\Phi(1,1,\ldots,1)\vee(p_1^0\vee p_2^0\vee\ldots\vee p_n^0)$$

zwanej koniunkcyjną postacią normalną.

PRZYKŁAD. Na podstawie twierdzenia 2 koniunkcyjna postać normalna formuły logicznej Φ będzie się składała tylko z takich alternatyw elementarnych $p^{1-c_1}\vee q^{1-c_2}\vee r^{1-c_3}$, dla których $\Phi(c_1,c_2,c_3)=1$, gdyż $1\vee p=1$ i inne alternatywy elementarne zostaną zredukowane do wartości logicznej 1. Czyli dla naszego przykładu alternatywami elementarnymi będą

$$p^{1-0}\vee q^{1-0}\vee r^{1-1}=p\vee q\vee\sim r,$$
$$p^{1-0}\vee q^{1-1}\vee r^{1-0}=p\vee\sim q\vee r,$$
$$p^{1-0}\vee q^{1-1}\vee r^{1-1}=p\vee\sim q\vee\sim r,$$
$$p^{1-1}\vee q^{1-1}\vee r^{1-1}=\sim p\vee\sim q\vee\sim r.$$

A więc koniunkcyjna postać normalna formuły będzie następująca

(4) $(p\vee q\vee\sim r)\,\&\,(p\vee\sim q\vee r)\,\&\,(p\vee\sim q\vee\sim r)\,\&\,\&\,(\sim p\vee\sim q\vee\sim r).$

Streszczenie

Każdą formułę rachunku zdań można przedstawić w dyzjunkcyjnej postaci normalnej bądź koniunkcyjnej postaci normalnej. Dyzjunkcyjna postać normalna jest sumą iloczynów wszystkich zmiennych zanegowanych lub nie, koniunkcyjna zaś postać normalna jest iloczynem sum wszystkich zmiennych zanegowanych lub nie.

Zadania

1. Udowodnić, że różnych formuł n zmiennych zapisanych w postaci normalnej jest 2^{2^n}.

2. Sprowadzić do dyzjunkcyjnej postaci normalnej formułę

$$\sim(p\,\&\,q\vee\sim r\,\&\,p)$$

a) metodą opisaną w tym paragrafie polegającą na ułożeniu tabelki;
b) za pomocą przekształceń formuły takich jak opisane w § 10.

3. Przedstawić w koniunkcyjnej postaci normalnej formułę:

$$(p \vee \sim r) \,\&\, q$$

a) metodą opisaną w tym paragrafie;

b) za pomocą przekształceń takich jak podane w § 10.

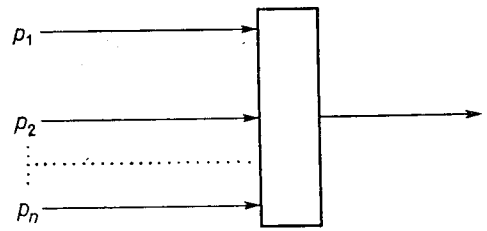
4. a) Udowodnić, że formuła n zmiennych jest tautologią wtedy i tylko wtedy, gdy dyzjunkcyjna postać normalna zawiera dokładnie 2^n składników.

b) Udowodnić, że formuła n zmiennych jest tautologią wtedy i tylko wtedy, gdy koniunkcyjna postać normalna ma dokładnie 2^n czynników.

§ 13. ELEKTRONOWA INTERPRETACJA SPÓJNIKÓW ZDANIOWYCH

W związku z rozwojem maszyn matematycznych rachunek zdań znalazł ciekawe zastosowanie techniczne. Tabliczki opisujące wartości logiczne zdań połączonych spójnikiem w zależności od wartości logicznych zdań składowych, można również interpretować jako opis działania pewnych układów elektronowych.

Rozpatrzmy układ, który ma wejścia $p_1, p_2, \ldots, p_n$ oraz jedno wyjście. Układ taki wygodnie jest oznaczać graficznie w następujący sposób:



Przyjmijmy, że każde z wejść lub wyjść układu może znajdować się w jednym z dwu możliwych stanów, które oznaczmy przez 0 i 1. Stanami wejść i wyjść układu mogą być np.

0 — brak impulsu,

1 — obecność impulsu,

bądź też

0 — obecność napięcia v_1,

1 — obecność napięcia v_2 $(v_1 \neq v_2)$.

czy też jakiekolwiek inne dwa rozróżnialne stany fizyczne. Działanie takiego układu jest więc jednoznacznie określone za pomocą tablicy podającej stan wyjścia w zależności od stanów wejść. Przyjmijmy ponadto, że układy takie działają bezczasowo, tzn. stan wyjścia nie zależy od czasu, a jedynie od aktualnych stanów wejść. Układy takie są nazywane *sieciami logicznymi*. Jeżeli działanie sieci jest opisane dwuwartościową tablicą T, to mówimy, że *układ realizuje tablicę* T.

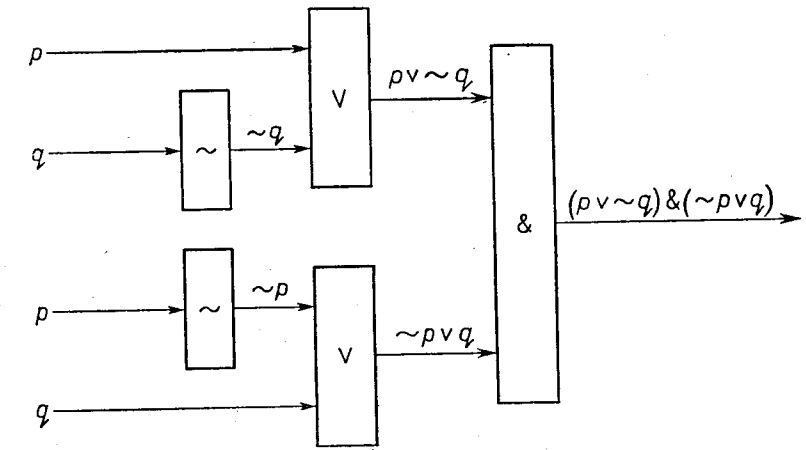
Podstawowe zadanie teorii sieci logicznych jest następujące: Zadane są układy (sieci) realizujące tablice: sumy logicznej, iloczynu logicznego i negacji — zwane *układami* (albo *sieciami*) *podstawowymi*, oraz dana jest dwuwartościowa tablica T. Zbudować sieć składającą się wyłącznie z układów podstawowych — realizującą tablicę T.

Jest to tzw. *problem syntezy sieci logicznych*. Problem ten sprowadza się do podania formuły rachunku zdań odpowiadającej danej tablicy T.

Jak wiemy z poprzedniego paragrafu zadanie to możemy rozwiązać, podając formułę w postaci normalnej dyzjunkcyjnej bądź koniunkcyjnej. Każda zaś formuła rachunku zdań może być w prosty sposób interpretowana jako schemat połączeń elementów podstawowych. Na przykład formule

$$(p \vee \sim q) \,\&\, (\sim p \vee q)$$

odpowiadać będzie schemat połączeń



Mając formułę w postaci normalnej, możemy od razu narysować sieć realizującą zadaną tablicę.

Posługując się prawami logicznymi, możemy formułę w postaci normalnej przekształcać, otrzymując równoważne jej formuły. Tak więc, rozwiązanie zadania syntezy nie jest jednoznaczne. Każde takie zadanie ma nieskończenie wiele rozwiązań. W związku z powyższym, drugim bardzo istotnym problemem teorii sieci logicznych jest znalezienie takiego rozwiązania, które nie tylko realizuje tablicę T, lecz spełnia inne dodatkowe warunki. W ten sposób zakres możliwych rozwiązań jest znacznie zawężony. Głównie chodzi tu o zbudowanie sieci realizującej zadaną tablicę za pomocą minimalnej liczby elementów podstawowych. Jest to tzw. *problem minimalizacji sieci* i sprowadza się on do znalezienia formuły zawierającej minimalną liczbę spójników zdaniowych, gdyż każdemu spójnikowi odpowiada jeden układ podstawowy(1).

Na temat minimalizacji sieci logicznych istnieje bardzo duża literatura, jednakże do tej pory problem ten nie został jeszcze rozwiązany w całej ogólności, a znane są dopiero rozwiązania w niektórych dość szczególnych przypadkach.

Inny problem związany z teorią sieci logicznych polega na analizie zadanej sieci logicznej, tj. na podaniu formuły, gdy zadana jest sieć. To zadanie jest bardzo łatwe i rozwiązanie jego nie sprawia trudności.

Jeszcze inne zadanie teorii sieci. Zbudować sieć realizującą zadaną tablicę logiczną, gdy zadane są układy realizujące funktory p/q albo $p\backslash q$. Również i w tym przypadku chodzi o podanie odpowiedniej postaci normalnej dla spójników „/" oraz „\", a także tautologii pozwalających na łatwe przekształcanie formuł zbudowanych ze zmiennych zdaniowych oraz funktorów „/" bądź „\", a także problem minimalizacji dla tych funktorów.

W związku z powyższymi zagadnieniami powstało szereg węższych problemów, bardziej specjalnych, jak np. analiza i synteza sieci z uwzględnieniem czasu, albo sieci zawierających sprzężenia zwrotne. Zbudowano do tej pory wiele rachunków, przypominających nieco rachunek zdań, służących do opisu działania sieci nerwowych i elektrycznych. Zastosowania techniczne i biologiczne rachunku zdań postawiły przed logiką szereg nowych problemów. Większość z nich nie została jeszcze do tej pory w zadowalający sposób rozwiązana.

(1) Takie postawienie problemu minimalizacji nie jest zbyt ścisłe, jednakże oddaje on z grubsza idee upraszczania sieci logicznych.

Zagadnienia syntezy i analizy sieci poruszone w tym paragrafie są wyczerpująco omówione w następujących podręcznikach: Millera, Kobryńskiego i Trachtenbrota, Mc Cluskey'a, A. W. Mostowskiego.

Streszczenie

Tabliczki spójników zdaniowych można również interpretować jako opis działania pewnych układów elektronowych. Formuły można wtedy uważać za liniowy zapis schematów sieci składających się z układów realizujących wybrane spójniki logiczne. Prawa logiczne można wtedy interpretować jako prawa upraszczania schematów sieci elektronicznych.

§ 14. AKSJOMATYCZNE UJĘCIE RACHUNKU ZDAŃ

Podane sformułowanie rachunku zdań miało charakter intuicyjny. Powiedzieliśmy jakie formuły są poprawnie zbudowane i wyróżniliśmy spośród nich te, które są twierdzeniami rachunku zdań — nazywając je zwyczajowo tautologiami. Rachunek zdań można również określić aksjomatycznie, Istnieje bardzo wiele różnych ujęć aksjomatycznych rachunku zdań. Przykładowo podamy w tym paragrafie trzy proste układy aksjomatów.

Poprzednio, jak pamiętamy, sprawdzenie prawdziwości twierdzenia rachunku zdań polegało na wykonywaniu prostych działań na wartościach logicznych 0 i 1. Przypominało ono wykonywanie rachunków arytmetycznych. W metodzie aksjomatycznej natomiast, aby sprawdzić czy dane wyrażenie tego rachunku jest twierdzeniem należy je wyprowadzić z ustalonych aksjomatów za pomocą przyjętych reguł dedukcji. We wszystkich trzech podanych układach aksjomatów będą obowiązywały dwie reguły wnioskowania: reguła podstawiania oraz reguła odrywania.

Reguła podstawiania mówi, że jeżeli za dowolną zmienną zdaniową w tautologii rachunku zdań podstawimy dowolną formułę rachunku zdań,

to otrzymana w ten sposób formuła jest również tautologią rachunku zdań. W szczególności możemy podstawić również zamiast formuły pojedynczą zmienną. Przy podstawianiu należy jedynie pamiętać, że podstawienia należy dokonać we wszystkich miejscach formuły, w których występuje dana zmienna. Na przykład, jeżeli w tautologii

$$p \Rightarrow (q \Rightarrow p)$$

zamiast p podstawimy formułę $p \vee q$, to otrzymana formuła

$$(p \vee q) \Rightarrow \big(q \Rightarrow (p \vee q)\big)$$

jest również tautologią logiczną, co łatwo sprawdzić metodą zero jedynkową.

Reguła odrywania pozwala na podstawie przesłanek p oraz $p \Rightarrow q$ przyjąć jako wniosek q, co zapiszemy symbolicznie

$$\frac{p,\ p \Rightarrow q}{q}.$$

Na przykład weźmy dwie tautologie

(1) $$(\sim p \Rightarrow p) \Rightarrow p,$$

(2) $$(q \Rightarrow p) \Rightarrow (\sim p \Rightarrow \sim q).$$

Podstawmy w tautologii (2) zamiast zmiennej q formułę

(3) $$\sim p \Rightarrow p.$$

Otrzymamy w ten sposób formułę

(4) $$\big((\sim p \Rightarrow p) \Rightarrow p\big) \Rightarrow \big(\sim p \Rightarrow \sim (\sim p \Rightarrow p)\big).$$

Stosując do tautologii (1) i (4) regułę odrywania, otrzymamy tautologię

(5) $$\sim p \Rightarrow \sim (\sim p \Rightarrow p).$$

Rozumowanie to możemy w skrócie zapisać w postaci:

$$\frac{(\sim p \Rightarrow p) \Rightarrow p,\ \big((\sim p \Rightarrow p) \Rightarrow p\big) \Rightarrow \big(\sim p \Rightarrow \sim(\sim p \Rightarrow p)\big)}{\sim p \Rightarrow \big(\sim(\sim p \Rightarrow p)\big)}$$

Obie te reguły wnioskowania będą obowiązywały dla wszystkich trzech dalej podanych systemów rachunku zdań.

PRZYKŁAD 1. Jako układ aksjomatów rachunku zdań możemy przyjąć następujące trzy aksjomaty:

S_1. $(\sim p \Rightarrow p) = p$,

S_2. $p \Rightarrow (\sim p \Rightarrow q)$,

S_3. $(p \Rightarrow q) \Rightarrow \big((q \Rightarrow r) \Rightarrow (p \Rightarrow r)\big)$.

Zaletą tego układu jest to, że zawiera on niewielką liczbę aksjomatów i tylko dwa spójniki pierwotne: implikację i negację. Alternatywę, $p \vee q$, definiujemy jako skrót zapisu formuły $\sim p \Rightarrow q$, koniunkcję zaś, $p \,\&\, q$, jako skrót zapisu formuły $\sim (p \Rightarrow \sim q)$, równoważność natomiast, $p \equiv q$, jako skrót zapisu formuły

$$(p \Rightarrow q)\ \&\ (q \Rightarrow p).$$

Jednak dowodzenie twierdzeń w oparciu o tak ubogi system aksjomatów jest niewygodne. Udowodnienie nawet prostej tautologii jest bardzo kłopotliwe. Por. zadanie 2 z tego paragrafu.

PRZYKŁAD 2. Mniej kłopotliwy w użyciu jest układ aksjomatów historycznie pochodzący od Whiteheda i Russella [14], zmodyfikowany przez Bernaysa [1].

Zawiera on dwa spójniki: alternatywę i negację. Dla skrócenia zapisu formuł logicznych piszemy

$$p \Rightarrow q \text{ zamiast } \sim p \vee q.$$

Regułami dowodzenia są opisane już reguły: odrywania (przypominamy, że reguła ta wyrażona jest w terminie implikacji) i podstawiania. W naszym przypadku reguła ta prowadzi od p i $\sim p \vee q$ do q.

System zawiera cztery aksjomaty:

S_1^*. $(p \vee p) \Rightarrow p$,

S_2^*. $p \Rightarrow (p \vee q)$,

S_3^*. $(p \vee q) \Rightarrow (q \vee p)$,

S_4^*. $(p \Rightarrow q) \Rightarrow \big((r \vee p) \Rightarrow (r \vee q)\big)$.

Przy takim układzie aksjomatów łatwo określić najczęściej używane spójniki zdaniowe.

Implikacja „⇒" została już określona jako prawidło skrótu formuły $\sim p \vee q$, koniunkcję $p \,\&\, q$ określamy jako skrót formuły $\sim(\sim p \vee \sim q)$, równoważność zaś $p \equiv q$ jako skrót formuły $(p \Rightarrow q) \,\&\, (q \Rightarrow p)$.

Wyprowadzenie znanych tautologii nie jest w tym systemie aksjomatów zbyt trudne. Systematyczny wywód bardziej znanych i potrzebnych tautologii z podanego systemu aksjomatów znajduje się w książce Hilberta i Ackermana [5].

Przykłady dowodzenia twierdzeń w tym systemie aksjomatów podane są w zadaniu 4 tego paragrafu.

PRZYKŁAD 3. Możliwy również i dość użyteczny jest system oparty o wszystkie najczęściej używane spójniki $\sim$, $\vee$, $\&$, $\Rightarrow$, $\equiv$. System taki musi z konieczności zawierać wiele aksjomatów, gdyż trzeba opisać własności wielu spójników. Na przykład system poniżej podany zawiera ich piętnaście:

S'_1. $p \Rightarrow (q \Rightarrow p)$,

S'_2. $\big(p \Rightarrow (p \Rightarrow q)\big) \Rightarrow (p \Rightarrow q)$,

S'_3. $(p \Rightarrow q) \Rightarrow \big((q \Rightarrow r) \Rightarrow (p \Rightarrow r)\big)$,

S'_4. $(p \,\&\, q) \Rightarrow p$,

S'_5. $(p \,\&\, q) \Rightarrow q$,

S'_6. $(p \Rightarrow q) \Rightarrow \Big((p \Rightarrow r) \Rightarrow \big(p \Rightarrow (q \,\&\, r)\big)\Big)$,

S'_7. $p \Rightarrow (p \vee q)$,

S'_8. $q \Rightarrow (p \vee q)$,

S'_9. $(p \Rightarrow r) \Rightarrow \Big((q \Rightarrow r) \Rightarrow \big((p \vee q) \Rightarrow r\big)\Big)$,

S'_{10}. $(p \equiv q) \Rightarrow (p \Rightarrow q)$,

S'_{11}. $(p \equiv q) \Rightarrow (q \Rightarrow p)$,

S'_{12}. $(p \Rightarrow q) \Rightarrow \big((q \Rightarrow p) \Rightarrow (p \equiv q)\big)$,

S'_{13}. $(p \Rightarrow q) \Rightarrow (\sim q \Rightarrow \sim p)$,

S'_{14}. $p \Rightarrow \sim(\sim p)$,

S'_{15}. $\sim(\sim p) \equiv p$.

W metodzie aksjomatycznej sprawdzenie prawdziwości tautologii logicznej polega na wyprowadzeniu jej z aksjomatów. Spróbujmy więc wyprowadzić prostą tautologię

(6) $\qquad p \Rightarrow p$

z trzeciego układu aksjomatów:

1. Aksjomat S'_1: $\quad p \Rightarrow (q \Rightarrow p)$;
2. podstawiamy w wierszu 1, p zamiast q: $\quad p \Rightarrow (p \Rightarrow p)$;
3. aksjomat S'_2: $\quad \big(p \Rightarrow (p \Rightarrow q)\big) \Rightarrow (p \Rightarrow q)$;
4. podstawiamy w wierszu 3, p zamiast q: $\quad \big(p \Rightarrow (p \Rightarrow p)\big) \Rightarrow (p \Rightarrow p)$;
5. do wierszy 2 i 4 stosujemy regułę odrywania: $\quad p \Rightarrow p$.

Streszczenie

Rachunek zdań można również określić na drodze aksjomatycznej. Istnieje wiele równoważnych sformułowań aksjomatycznych rachunku zdań. Jako reguły wnioskowania w tych sformułowaniach przyjmuje się na ogół regułę podstawiania oraz regułę odrywania.

Zadania

1. a) Udowodnić metodą sprawdzania zero-jedynkowego, że jeżeli formuły $f(p, q, r, \ldots)$ oraz $f(p, q, r, \ldots) \Rightarrow g(p, q, r, \ldots)$ są tautologiami, to tautologią jest również formuła $g(p, q, r, \ldots)$.

b) Udowodnić, że stosując do tautologii podstawienie otrzymamy znowu tautologię.

2. Podstawiając w S_3 zamiast zmiennej q formułę $\sim p \Rightarrow p$ oraz zamiast zmiennej r zmienną p, wyprowadzić z aksjomatów S_1-S_3 tautologię $p \Rightarrow p$.

3. a) Zapisać aksjomaty S^*_1-S^*_4 z przykładu 2 za pomocą alternatywy „$\vee$" i negacji „$\sim$", pisząc zamiast formuły $p \Rightarrow q$ formułę $\sim p \vee q$.

b) Wyrazić regułę odrywania za pomocą alternatywy „$\vee$" i negacji „$\sim$".

4. a) Wypisać formułę:

(*) $\qquad (p \Rightarrow q) \Rightarrow (r \Rightarrow p) \Rightarrow (r \Rightarrow p)$

i formułę będącą podstawieniem S^*_4

$$(p \Rightarrow q) \Rightarrow (\sim r \vee p) \Rightarrow (\sim r \vee q)$$

za pomocą tylko alternatywy „$\vee$" i negacji „$\sim$". Stwierdzić, że obie formuły są skrótem zapisu jednej i tej samej funkcji zdaniowej. Wywnioskować stąd, że (*) jest wnioskiem z aksjomatów S^*_1-S^*_4 podanych w przykładzie 2.

b) Podstawiając do (*) zamiast p formułę $q \vee q$, zamiast r zaś formułę q, udowodnić, że

**) $$q \Rightarrow q$$

jest twierdzeniem. Formuła (**) jest skróconym zapisem formuły

***) $$\sim q \vee q.$$

Formuła ta jest więc również twierdzeniem.

c) Na podstawie aksjomatów S_1^*-S_4^* i udowodnionych formuł, udowodnić następujące formuły:

$$p \Rightarrow \sim (\sim p),$$

$$(p \Rightarrow q) \Rightarrow (\sim q \Rightarrow \sim p).$$

Rozdział 3

RACHUNEK KWANTYFIKATORÓW

W poprzednim rozdziale określiliśmy strukturę pewnej klasy zdań występujących w teoriach matematycznych oraz podaliśmy prawa rządzące prawdziwością tych zdań. Prawa te dotyczyły właściwie zdań nie tylko o treści matematycznej, lecz dowolnych zdań o strukturze poprawnych formuł, tzn. zdań otrzymanych ze składowych zdań oznajmiających, połączonych odpowiednio spójnikami zdaniowymi. Rachunek zdań nie stanowi jednak całego języka stosowanego w rozumowaniach matematycznych.

Dopiero w końcu XIX wieku zaczął powstawać nowy fragment logiki — rachunek kwantyfikatorów — który łącznie z rachunkiem zdań obejmuje cały język wszelkich teorii matematycznych oraz prawa posługiwania się tym językiem. Język rachunku zdań i rachunku kwantyfikatorów pozwala na wyrażanie myśli dowolnych teorii matematycznych i sprawdzanie ich prawdziwości za pomocą formalnego postępowania dowodowego. Całą obecną wiedzę matematyczną można by wyrazić, posługując się językiem rachunku zdań i rachunku kwantyfikatorów. Jednakże jak wiadomo, za wszelką uniwersalność zawsze czymś się płaci. W tym przypadku idzie o to, że postępowanie takie jest bardzo uciążliwe i rozwlekłe. Przyczyny, dla których musimy się jednak decydować na takie postępowanie, zostały omówione w § 4, mówiącym o formalizacji matematyki. Ujemne strony sprowadzenia matematyki do rozważań nad rachunkiem zdań i rachunkiem kwantyfikatorów zostaną omówione w rozdziale IV.

§ 15. ZDANIA I FUNKCJE ZDANIOWE

Struktura zdań występujących w teoriach matematycznych jest w porównaniu ze strukturą zdań języka potocznego, stosunkowo uboga. Nato-

miast w języku matematyki występuje pojęcie, które w języku potocznym jest nie znane — jest to pojęcie *funkcji zdaniowej*. Tak więc język matematyki jakkolwiek zazębia się z językiem potocznym, to jednak w istocie różni się dość zasadniczo od niego.

Przykładem funkcji zdaniowej są wyrażenia:

(1) x jest większe od 10,

(2) x jest mniejsze od y,

(3) x jest podzielne przez 2,

(4) x jest równe 0.

Funkcje zdaniowe nazywane są też *predykatami*.

Żadne z wyrażeń (1)-(4) nie jest zdaniem oznajmiającym. Zdania oznajmiające bowiem wyrażają sądy prawdziwe albo fałszywe. O wyrażeniach (1)-(4) nie możemy powiedzieć, że są one prawdziwe bądź fałszywe. Dopiero podstawiając za argumenty x i y wartości liczbowe otrzymamy z tych wyrażeń zdania. Na przykład, jeżeli w wyrażeniu (1) za x podstawimy liczbę 5, to otrzymamy zdanie

(5) 5 jest większe od 10.

które jest oczywiście fałszywe, natomiast podstawiając za x liczbę 20, otrzymamy zdanie prawdziwe

(6) 20 jest większe od 10.

Funkcjami zdaniowymi są również wyrażenia:

(7) x jest żoną y,

(8) x ma wysokość y metrów,

(9) x jest synem y

itp. Z funkcji tych otrzymamy zdania, jeżeli na miejsce x i y podstawimy odpowiednie słowa. Na przykład jeżeli w predykacie (7) za x podstawimy Ina, a za y — Andrzej, to otrzymamy zdanie

(10) Ina jest żoną Andrzeja.

Zauważmy na marginesie, że o ile prawdziwość zdań (5) bądź (6) może być określona bezpośrednio, na podstawie rozumienia sensu słowa „większy", to z określeniem prawdziwości zdania (10) są większe kłopoty. Musimy bowiem wiedzieć o którą i Inę i Andrzeja chodzi w naszym zdaniu i wiedzieć czy oni są małżeństwem. W zdaniach, które będziemy rozważać zachodzi pierwsza wymieniona sytuacja, tj. podstawiając w predykacie za zmienne odpowiednie słowa, możemy jednoznacznie określić prawdziwość otrzymanego zdania, niezależnie od żadnych warunków ubocznych.

W językach teorii matematycznych mamy więc trzy rodzaje pojęć:

1. zdania,
2. schematy zdań (albo formuły rachunku zdań),
3. funkcje zdaniowe (albo predykaty).

Dokładne zdanie sobie sprawy z różnic między tymi trzema pojęciami jest nieodzowne dla dokładnego rozumienia języka matematyki. Dlatego w dalszym ciągu poświęcimy im jeszcze nieco uwagi.

Zdania określają pewne stosunki, zależności, czy jak też mówimy niekiedy relacje, między przedmiotami. Na przykład

5 jest większe od 2

określa pewną zależność między dwoma obiektami, liczbami 5 i 2. Podobnie zdanie

Ina jest żoną Andrzeja

opisuje pewną zależność między dwoma obiektami Iną i Andrzejem.

Formuły logiczne są schematami zdań poprawnych. Stanowią one jak gdyby gramatykę logiki, pokazując, jak należy posługiwać się poprawnie spójnikami logicznymi.

Funkcje zdaniowe natomiast opisują związki w jakiejś klasie przedmiotów. Podstawiając np. w zdaniu

x jest większe od y

za x i y różne liczby naturalne, otrzymamy związek w klasie liczb, tych właśnie, które możemy podstawiać za x i za y.

Do spraw tych wrócimy jeszcze w dalszych paragrafach.

W języku matematyki zamiast słów języka potocznego używa się specjalnych symboli. Nie piszemy np.

5 jest większe od 2,

lecz zamiast zwrotu „jest większe od" używamy jednego symbolu „$>$". Tak więc powyższe zdanie zapisujemy krótko

$$5 > 2.$$

Podobnie piszemy inne zdania

$$2 \neq 3, \quad 1 = 1, \quad 2+5 = 7.$$

Dla każdego rodzaju stosunków zachodzących między interesującymi nas przedmiotami można wprowadzić specjalne symbole tak, że wszelkie zdania możemy zapisywać bez potrzeby używania słów z języka potocznego. Podobnie można oczywiście zapisywać zdania złożone, jak np.

$$[(1 = 0) \,\&\, (2 \neq 3)] \equiv [\sim (2 > 7) \vee (2+5 = 7)].$$

Podobnie predykaty, piszemy w matematyce na ogół bez używania słów języka potocznego.

$$x > 2, \quad x > y, \quad x \neq y,$$

$$x+y = 5, \quad x \cdot y = z.$$

Predykaty będziemy oznaczali

$$P(x), \quad Q(x, y), \quad R(x, y, z).$$

Oczywiście zamiast liter P, Q, R będziemy też stosować inne duże litery. (Jeżeli nie będą nas interesowały zmienne, to predykaty będziemy oznaczali dowolnymi dużymi literami alfabetu łacińskiego). A więc $P(x, y)$ może oznaczać zarówno predykat $x > y$, $x \neq y$ lub $x+y = 2$, jak też jakikolwiek inny predykat o dwu zmiennych.

Predykaty, podobnie jak zdania, możemy łączyć spójnikami, o których była mowa w rozdziale poprzednim, otrzymując w ten sposób nowe predykaty. Na przykład predykat $Q(x, y, z)$ może mieć postać

$$[(x > 2) \vee (x \neq y)] \,\&\, (x+y = z)$$

lub krótko

$$(P \vee R) \,\&\, S, \tag{11}$$

gdzie

$$P(x) = (x > 2), \; R(x, y) = (x \neq y) \text{ oraz } S(x, y, z) = (x+y = z).$$

Zwróćmy uwagę, że chociaż formuła (11) przypomina formułę rachunku zdań, nie jest ona schematem zdania a schematem predykatu. Zaznaczyliśmy to, pisząc argumenty dużymi literami, które będziemy nazywać *zmiennymi predykatywnymi*. (W rachunku zdań jako argumentów używaliśmy małych liter, które były zmiennymi zdaniowymi, tj. na ich miejsce mogliśmy podstawiać zdania).

Symbole takie jak, 0, 5, $+$, $=$, $>$ nazywamy *stałymi*. Wszystkie one mają określone znaczenie i oznaczają bądź nazwy liczb (czy innych rozważanych w teorii obiektów), bądź działania jak np. symbol „$+$", bądź też relacje między obiektami, jak np. symbole „$=$", „$\neq$", „$>$". Symbole oznaczające stosunki między przedmiotami będziemy nazywać *stałymi predykatywnymi*. Wszystkie stałe mają również swoje nazwy w gramatyce, jak np. rzeczownik, czasownik itp.

Zmienna w predykacie natomiast nie ma żadnego określonego znaczenia, nie jest nazwą ani przedmiotu, ani operacji, ani też stosunku. Jeżeli podstawiając w predykacie za zmienne odpowiednie stałe otrzymamy zdanie prawdziwe, to mówimy, że stałe te spełniają predykat (lub funkcję zdaniową). W przypadku przeciwnym funkcja zdaniowa nie jest spełniona.

Zauważmy jeszcze, że wyrażenia takie jak

$$x+y, \quad x^2-y,$$

nie są funkcjami zdaniowymi. Podstawiając bowiem w nich w miejsce zmiennych odpowiednie stałe, nie otrzymamy zdań, tylko wyrażenia takie jak

$$5+2, \quad 6^2-4, \quad \text{itp.}$$

Wyrażenia te nie zawierają orzeczenia (symbolu predykatywnego) nie są więc one zdaniami. Są to po prostu nazwy pewnych przedmiotów.

Streszczenie

Wyrażenia mające budowę zdań i zawierające zmienne, za które można podstawiać nazwy przedmiotów z ustalonego zbioru, są nazywane funkcjami zdaniowymi albo predykatami. Jeżeli w funkcji zdaniowej na miejsce zmiennych podstawimy odpowiednie nazwy, to otrzymamy zdanie prawdziwe lub fałszywe. Predykaty nie są natomiast ani prawdziwe, ani fałszywe, nie wyrażają bowiem żadnych sądów.

Fukcje zdaniowe można ze sobą łączyć spójnikami logicznymi, według takich samych zasad jak zdania, otrzymując w ten sposób nowe funkcje zdaniowe.

§ 16. FUNKCJE ZDANIOWE I ZBIORY

Za zmienne w funkcjach zdaniowych nie możemy wstawiać nazw dowolnych przedmiotów. Zbiór przedmiotów, których nazwy możemy podstawiać, musi być ściśle określony. Na przykład jeżeli w funkcji zdaniowej

$$x \text{ jest żoną } y$$

za x i y podstawimy liczby naturalne, to otrzymamy oczywiście zdanie bez sensu

$$5 \text{ jest żoną } 2.$$

A więc zbiorem, którego nazwy elementów możemy podstawiać w powyższym predykacie za x, jest zbiór kobiet, a zbiorem, którego nazwy elementów możemy podstawiać za y, jest zbiór mężczyzn.

Z każdą zmienną w predykacie jest więc związany pewien zbiór przedmiotów. Jeżeli funkcja zdaniowa jest jednoargumentowa, tzn. występuje w nim jedna zmienna, to zbiór przedmiotów które można wstawiać za zmienną w predykacie i otrzymywać w wyniku zdanie sensowne nazywamy *dziedziną predykatu*. Jeżeli predykat jest wielo-argumentowy, to zbiory odpowiadające każdej zmiennej nazywamy *dziedziną pierwszą*, *dziedziną drugą* itd. W szczególnym przypadku, jeżeli predykat jest dwuargumentowy, to dziedzinę pierwszą nazywamy po prostu *dziedziną predykatu*, a dziedzinę drugą — *przeciwdziedziną*. O zmiennych mówimy wtedy, że przyjmują wartości z odpowiedniej dziedziny, bądź też, że zmienne przebiegają dany zbiór przedmiotów. Na przykład dziedziną funkcji zdaniowych

$$P(x) = (x > 2), \tag{1}$$

$$Q(x) = (x = 6), \tag{2}$$

$$R(x) = (x \neq 0) \tag{3}$$

jest zbiór liczb naturalnych. Dla funkcji zdaniowych o dwu argumentach

$$P(x, y) = (x = y), \tag{4}$$

$$Q(x, y) = (x+5 = y), \tag{5}$$

$$R(x, y) = (x > y) \tag{6}$$

dziedziną oraz przeciwdziedziną są zbiory liczb naturalnych(1).

Natomiast dla funkcji zdaniowej

$$\text{pan } x \text{ waży } y \text{ kilogramów}$$

dziedziną jest zbiór wszystkich mężczyzn a przeciwdziedziną zbiór liczb 0, 1, 2, ..., 150 (przyjmujemy tu, że ważymy z dokładnością do 1 kg i że nie ma ludzi o wadze większej niż 150 kg).

Zbiór przedmiotów, dla których funkcja zdaniowa jednej zmiennej przechodzi w zdanie prawdziwe nazywamy *zbiorem rozwiązań* danej funkcji zdaniowej, bądź krótko *rozwiązaniem* funkcji zdaniowej. Na przykład dla predykatów (1), (2), (3) rozwiązaniami będą odpowiednio następujące zbiory liczb naturalnych.

$$Z_1 = \{3, 4, 5, \ldots\},$$

$$Z_2 = \{6\},$$

$$Z_3 = \{1, 2, 3, \ldots\}.$$

Tutaj $\{3, 4, 5, \ldots\}$ oznacza zbiór złożony z liczb 3, 4, 5, ..., $\{6\}$ zaś oznacza zbiór, którego jedynym elementem jest liczba 6. Zbiory Z_1 i Z_3 są nieskończone, natomiast zbiór Z_2 jest skończony i zawiera tylko jeden element — liczbę 6.

(1) Przyjmujemy tutaj, że liczby naturalne są to liczby 0, 1, 2, 3, ..., tzn. zaliczamy zero do liczb naturalnych. Oczywiście jako dziedzinę i przeciwdziedzinę moglibyśmy przyjąć tutaj zbiory liczb rzeczywistych, wymiernych czy jakichkolwiek przedmiotów dla których są określone operacje i relacje występujące w tych funkcjach zdaniowych. Otrzymalibyśmy wtedy właściwie inny predykat, o innym zakresie zmienności.

O elementach zbiorów Z_1, Z_2, Z_3 mówimy, że spełniają funkcje zdaniowe odpowiednio (1), (2) i (3). Funkcje zdaniowe określone dla liczb naturalnych nazywamy *funkcjami spełnialnymi*, gdy istnieją liczby naturalne spełniające te funkcje.

Rozważmy jeszcze funkcję zdaniową $S(x) = (x = x)$ określoną w zbiorze liczb naturalnych. Funkcja ta jest spełniana przez każdą liczbę naturalną x. Podstawiając za x dowolna liczbę, otrzymamy zdanie prawdziwe. Taką *funkcję zdaniową* będziemy nazywali *prawdziwą*. Jej rozwiązanie stanowi zbiór

$$Z_4 = \{0, 1, 2, \ldots\},$$

składający się ze wszystkich liczb naturalnych.

Funkcje zdaniowe prawdziwe są oczywiście spełnialne. Nie każda funkcja zdaniowa jest jednak spełnialna. Na przykład negacja

$$\sim S(x) = (x \neq x)$$

funkcji prawdziwej $(x = x)$ jest niespełnialna, nie ma liczb naturalnych, które by spełniały tę funkcję. Zbiorem jej rozwiązań jest zbiór pusty nie zawierający żadnych elementów. Pierwszy element pary oznacza wartość zmiennej x, drugi zaś — wartość zmiennej y.

Streszczenie

Zbiór przedmiotów, których nazwy możemy podstawiać na miejsce zmiennych w predykacie nazywany jest dziedziną tego predykatu. Zbiór elementów, dla których predykat jest prawdziwy nazywamy rozwiązaniem predykatu. Funkcje zdaniowe, dla których zbiór rozwiązań jest nie pusty nazywamy spełnialnymi. Te spośród nich, dla których zbiór rozwiązań jest całą dziedziną nazywamy funkcjami zdaniowymi prawdziwymi.

Zadania

1. Znaleźć zbiór wszystkich par x, y liczb rzeczywistych (punktów płaszczyzny) dla których następujące predykaty są zdaniami prawdziwymi:

a) $x < y$; b) $\sim (x < y)$; c) $(x \leqslant y) \vee (y+1) = x$;

d) $x^2+y^2 = 1$; e) $2^2+y^2 < 1$.

2. W geometrii zbiór punktów płaszczyzny (x, y), dla których predykat $P(x, y)$ (gdzie zmienne x i y przebiegają liczby rzeczywiste) jest zdaniem prawdziwym nazywamy zwykle *miejscem geometrycznym*. W poprzednim zadaniu znajdowaliśmy przykłady miejsc geometrycznych, predykat zaś $P(x, y)$ był *własnością punktu* (x, y).

Znaleźć miejsce geometryczne dla następujących własności:

a) $x^2+y^2+1 \geqslant 0$; b) $x^2+y^2-1 < 0$;

c) $x = 7$; d) $(x = 2) \,\&\, (y = 3)$; e) $(x-2)^2+(x-3)^2 = 0$.

3. Wskazać dziedziny następującego predykatu $P(N, x)$:

pożyczyłem x złotych Janowi N

a) Jak może wyglądać „miejsce geometryczne" tego predykatu? Czy musi się składać z jednego Jana i określonej sumy złotych?

b) To samo dla predykatu $Q(N, X)$:

pożyczyłem X złotych od Jana N.

Czy oba predykaty mogą mieć te same miejsca geometryczne? i co by to znaczyło?

4. Zapisać symbolami matematycznymi $=, +, \cdot, <$ predykaty $P(x, y)$, gdzie x, y przebiegają liczby rzeczywiste, mające następujące miejsca geometryczne:

a) Koło o promieniu 1 i środku w początku układu współrzędnych.

b) Okrąg o promieniu 1 i takim samym środku.

c) Wnętrze kwadratu o środku w początku układu, boku równym 2, którego przekątne leżą na osiach układu. Jakich spójników logicznych musimy do tego użyć?

5. a) Udowodnić, że jeżeli predykaty $P(x)$ oraz $Q(x)$ są spełnialne, to predykat $P(x) \vee Q(x)$ jest spełnialny.

b) To samo udowodnić dla predykatu $P(x) \Rightarrow Q(x)$.

c) Podać przykład, w którym przy założeniach punktu a) predykat $P(x) \,\&\, Q(x)$ nie musi być spełnialny.

d) Kiedy predykat $\sim P(x)$ jest spełnialny?

§ 17. KWANTYFIKATORY

Rozpatrzmy następujące przykłady funkcji zdaniowych:

(1) $$P(x, y) = (x+y = y+x),$$

(2) $$Q(x, y) = (x = y),$$

(3) $$R(x) = (x = 4).$$

Przyjmijmy, że dziedzinami wszystkich tych funkcji zdaniowych jest zbiór liczb naturalnych.

Pierwsza funkcja zdaniowa $P(x, y)$ jest prawdziwa dla wszystkich liczb naturalnych. Mówiąc inaczej, jej rozwiązaniem jest zbiór dowolnych par liczb naturalnych. Możemy więc powiedzieć:

dla dowolnych liczb naturalnych x i y: $x+y = y+x$.

Funkcja zdaniowa $Q(x, y)$ natomiast nie jest prawdziwa dla dowolnych par liczb naturalnych, a tylko dla niektórych. Na przykład para liczb (2,6) nie spełnia tej funkcji zdaniowej, natomiast pary (2,2), (4,4) spełniają tę funkcję zdaniową. W tym przypadku powiedzielibyśmy:

dla niektórych liczb naturalnych x i y: $x = y$.

Trzecia wreszcie funkcja zdaniowa $R(x)$, jest prawdziwa tylko dla jednej liczby 4, dla wszystkich innych zaś liczb naturalnych jest ona fałszywa. Powiedzielibyśmy więc:

istnieje taka liczba naturalna x, że: $x = 4$.

Również zamiast zwrotu: „dla niektórych liczb naturalnych" możemy użyć zwrotu „istnieją takie liczby naturalne ..., że ...", pisząc:

istnieją takie liczby naturalne x i y, że: $x = y$.

Zwroty

dla każdego x,

istnieją takie x, że

odgrywają w języku matematyki zasadniczą rolę i łączenie ze spójnikami zdaniowymi, stałymi i zmiennymi pozwalają już na wypowiadanie każdej myśli matematycznej. Są one nazywane *kwantyfikatorami*. Pierwszy z nich nazywamy *kwantyfikatorem ogólnym* lub *dużym*, natomiast drugi kwantyfikator — *kwantyfikatorem szczegółowym, egzystencjalnym*, albo *małym*. Zamiast zwrotów „dla każdego x" oraz „istnieje takie x, że" używane są skróty podane w poniższej tabelce:

Dla każdego x	Ax	$\underset{x}{\forall}$	(x)	$\underset{x}{\Pi}$	$\underset{x}{\wedge}$	$\underset{x}{\cap}$
Istnieje takie x, że	Ex	$\exists x$	$\exists x$	$\underset{x}{\Sigma}$	$\underset{x}{\vee}$	$\underset{x}{\cup}$

My będziemy tu używali pierwszego oznaczenia, tj. symbolu Ax dla kwantyfikatora dużego i symbolu Ex dla kwantyfikatora małego.

Poprzednie przykłady funkcji zdaniowych poprzedzone kwantyfikatorami możemy teraz zapisać:

$$Ax\ Ay\ [x+y = y+x],$$

$$Ex\,Ey\,[x = y].$$

$$Ax\ [x = 4],$$

Zamiast pisać $AxAy$ będziemy często pisali krócej Axy i podobnie zamiast $ExEy$ będziemy pisali Exy. Dwa pierwsze wyrażenia zapiszemy więc

$$Axy[x+y = y+x],$$

$$Exy[x = y].$$

Dla uproszczenia nie będziemy również pisali wszystkich nawiasów w wyrażeniach zawierających kwantyfikatory a pozostawimy tylko te nawiasy, które są konieczne dla jednoznacznego odczytania wyrażenia. Przed funkcjami zdaniowymi nie zawierającymi spójników logicznych będziemy pisali kwantyfikatory w następujący sposób

$$Axy(x+y = y+x),$$

$$Exy(x = y),$$

$$Ex(x > y).$$

Zamiast

$$Ax\,[Ey\,(P(x, y))]$$

będziemy pisali

$$Ax\,Ey\,P(xy).$$

Podobnie dla funkcji wielu zmiennych.

Jeżeli zmienne przebiegają skończony zbiór przedmiotów, to kwantyfikatory rozumiemy po prostu jako sumę logiczną i iloczyn logiczny zdań otrzymanych z występującej pod nim funkcji zdaniowej, przez podstawienie kolejnych elementów dziedziny. Jeżeli więc dziedziną funkcji $P(x)$ jest zbiór n-elementowy $\{a_1, a_2, \ldots, a_n\}$, to kwantyfikatory rozumiemy w następujący sposób:

kwantyfikator $AxP(x)$ jako

$$P(a_1) \,\&\, P(a_2) \,\&\, \ldots \,\&\, P(a_n).$$

kwantyfikator zaś $ExP(x)$ jako

$$P(a_1) \vee P(a_2) \vee \ldots \vee P(a_n).$$

Kwantyfikatory można więc interpretować jako uogólnienie iloczynu i sumy logicznej na przypadek nieskończonej ilości zdań składowych.

Wyrażenie znajdujące się w nawiasie występującym bezpośrednio po kwantyfikatorze nazywamy *zasięgiem kwantyfikatora*. Na przykład w wyrażeniu

(4) $$Ay[x+x = y+x]$$

zasięgiem kwantyfikatora Ay jest wyrażenie

(5) $$x+y = y+x.$$

Natomiast w wyrażeniu

(6) $$AxAy[(x > y)] \,\&\, (x \neq y)$$

zasięgiem kwantyfikatora Ay jest wyrażenie

(7) $$x > y,$$

zasięgiem zaś kwantyfikatora Ax jest wyrażenie

(8) $$Ay[x > y].$$

Na przykład w wyrażeniu

$$Ax\{Ex[P(x)] \,\&\, R(x, y)\} \equiv Q(x)$$

zasięgiem kwantyfikatora Ex jest wyrażenie $P(x)$, a zasięgiem kwantyfikatora Ax jest wyrażenie

$$Ex[P(x)] \,\&\, R(x, y).$$

Mówimy, że *zmienna* występująca w wyrażeniu przy symbolu kwantyfikatora jest przez ten kwantyfikator *związana*, jeżeli znajduje się ona w zasięgu kwantyfikatora. W przeciwnym przypadku *zmienna* występująca w wyrażeniu jest *wolna*. Na przykład w wyrażeniu (6) zmienne x i y stojące po obu stronach znaku $>$ są związane, natomiast te same zmienne x i y stojące po obu stronach znaku $\neq$ są wolne, gdyż nie znajdują się one w zasięgu kwantyfikatorów Ax i Ay. Podobnie w wyrażeniu

$$Ay\{Ex[(x > y) \Rightarrow (x+y = \underline{z})] \,\&\, (\underline{x} = y)\}$$

podkreślone zmienne z oraz x są wolne, natomiast zmienne nie podkreślone są związane.

Streszczenie

Zwroty „dla każdego x" oraz „istnieje takie x, że" odgrywają podstawową rolę w wypowiadaniu myśli o treści matematycznej. Zwroty te nazywamy kwantyfikatorami i oznaczamy symbolami Ax oraz Ex. Kwantyfikatory wraz ze spójnikami logicznymi, symbolami operacji, nazwami przedmiotów oraz zmiennymi tworzą zasób pojęć stanowiących podstawę każdego języka matematycznego.

Kwantyfikatory wiążą zmienne znajdujące się w ich zasięgu. Ta sama zmienna może więc być w jednym miejscu związana w innym zaś wolna.

Zadania

1. Kwantyfikatory przebiegają zbiór skończony złożony z liczb 1, 2, 3, 4. Zapisać za pomocą spójników „$\vee$" oraz „&" następujące funkcje zdaniowe:

a) $Ax(x \leqslant y)$; b) $Ey\,Ax(x \leqslant y)$; c) $Ax\,Ey(x \leqslant y)$;

d) $Ey(y|x)$; e) $Ey\,Ax(y|x)$; f) $Ax\,Ey(y|x)$,

gdzie $u|v$ oznacza, że u dzieli v, czyli v jest krotnością u.

Udowodnić, że przy zadanym zakresie zmienności zmiennej x są to zdania prawdziwe lub funkcje zdaniowe prawdziwe, które przechodzą w zdania prawdziwe dla każdego podstawienia za zmienną liczb 1, 2, 3, 4.

2. a) Wskazać zakresy kwantyfikatora w następujących predykatach. Wyróżnić zmienne wolne i zmienne związane.

$$Ax\,[Ey\,[((x+y)^2 < z) \vee (u+x < v)] \vee (y+x = u)].$$

b) To samo wskazać dla predykatu:

$$Ax\,[Ey\,[((x+y)^2 < z) \vee (u+x < v)] \vee (z+x = u)].$$

c) Czy oba predykaty są naprawdę od siebie różne? Czy zmienną y związaną w b) można zastąpić w b) inną zmienną np. literą z?

3. Wyróżnić zmienne wolne i zmienne związane w następujących wyrażeniach matematycznych:

a) $\sum_{i=1}^{\infty} a_{ik}^n$, $\sum_{k=1}^{\infty}\sum_{i=1}^{\infty} a_{ik}^n$, $\sum_{n=1}^{10}\sum_{k=1}^{\infty}\sum_{i=1}^{\infty} a_{ik}^n$;

b) $\frac{1}{x^2+1}$, $\int_a^b \frac{1}{x^2+1}\,dx$, $\int_0^t \frac{1}{x^2+1}\,dx$, $\int_0^x f(x)\,dx$,

c) Które z tych wyrażeń przedstawiają funkcje, a które liczby?

§ 18. REGUŁY OPEROWANIA KWANTYFIKATORAMI

Jeżeli funkcję zdaniową poprzedzimy kwantyfikatorami w ten sposób, że żadna z występujących w niej zmiennych nie jest zmienną wolną, a wszystkie zmienne są związane, to otrzymane tak wyrażenie nie jest już funkcją zdaniową, na przykład

(1) $$AxAy(x+y = y+x),$$

(2) $$AxEy(x+y = y+x),$$

(3) $$AxAy(x > y),$$

(4) $$ExAy(x = y).$$

Każde z powyższych wyrażeń jest funkcją zdaniową zera zmiennych czyli zdaniem wyrażającym pewien sąd prawdziwy albo fałszywy. Rozumiejąc sens powyższych napisów łatwo stwierdzić, że (1) i (2) są zdaniami prawdziwymi, natomiast (3) i (4) — zdaniami fałszywymi. Kwantyfikatory pozwalają więc tworzyć z funkcji zdaniowych zdania. Stosując do funkcji zdaniowej n zmiennych, k razy kwantyfikatory, otrzymamy funkcję zdaniową n-k zmiennych w szczególności dla $n = k$, będzie to funkcja zdaniowa zera zmiennych, a więc zdanie.

Należy przy tym pamiętać, że zdania zawierające kwantyfikatory muszą mieć ściśle sprecyzowane zbiory przedmiotów (dziedziny), które mogą przebiegać zmienne występujące w zdaniu. Kwantyfikatory wyrażają bowiem, że jakaś własność zachodzi dla wszystkich, albo dla niektórych elementów tego zbioru.

Mamy więc dwie metody otrzymywania zdań z funkcji zdaniowych. Pierwsza z nich polega na podstawieniu w funkcji zdaniowej na wszystkie zmienne odpowiednich nazw, druga zaś — na poprzedzeniu funkcji zdaniowej odpowiednią ilością kwantyfikatorów, tak aby wszystkie zmienne wolne występujące w funkcji zdaniowej zostały związane przez kwantyfikatory. W języku potocznym zdań drugiego typu nie spotykamy. Podkreślamy więc raz jeszcze, że struktura języka matematyki odbiega znacznie od struktury języka potocznego. Język ten ma strukturę przystosowaną odpowiednio do wyrażania własności tworów badanych w matematyce, takich jak liczby, punkty, zbiory, funkcje itp. Język potoczny jest do tego celu wysoce niewygodny.

Jeżeli natomiast funkcję zdaniową dwóch zmiennych $P(x, y)$ poprzedzimy kwantyfikatorem, np. Ax, to otrzymane wyrażenie

$$AxP(x, y)$$

jest nadal funkcją zdaniową jednej zmiennej y. Zmienna x została bowiem związana i nie możemy za nią podstawiać żadnych nazw, gdyż otrzymalibyśmy napisy pozbawione sensu, jak np.:

$$A5P(5, y).$$

Powiedzenie „dla każdego 5" nie ma sensu, albowiem jest tylko jeden przedmiot oznaczony symbolem 5. W rezultacie w wyrażeniu

$$AxP(x, y) = Q(y)$$

istnieje tylko jedna zmienna y, za którą możemy podstawiać nazwy różnych przedmiotów z ustalonego zbioru. Funkcja ta zależy więc tylko od jednej zmiennej.

W ten sposób, stosując do funkcji zdaniowej o dowolnej liczbie zmiennych wolnych jeden kwantyfikator wiążący jedną zmienną występującą w tej funkcji, otrzymamy funkcję zdaniową, która ma jedną zmienną wolną mniej niż funkcja pierwotna. Stosując do tak otrzymanej funkcji jeszcze jeden kwantyfikator, otrzymamy funkcję zdaniową zawierającą jeszcze jedną zmienną wolną mniej. Postępując podobnie dalej, otrzymamy w rezultacie zdanie. Tak więc każdą funkcję zdaniową możemy w ten sposób przekształcić w zdanie.

Do funkcji zdaniowych możemy jeszcze stosować ważne operacje podstawienia zmiennych.

Rozpatrzmy następujący przykład funkcji zdaniowej

$$P(x, y) = (x < y) \vee (x^2+y^2 = 1)$$

dwóch zmiennych x i y. Za zmienną x możemy podstawić dowolną inną zmienną np. z. Wtedy otrzymamy funkcję zdaniową dwóch zmiennych z i y

$$P(z, y) = (z < y) \vee (z^2+y^2 = 1).$$

Podstawiając natomiast za zmienną x zmienną y otrzymamy również funkcję zdaniową, ale już jednej zmiennej

$$P(y) = P(y, y) = (y < y) \vee (y^2+y^2 = 1).$$

Operacja podstawiania zmiennej z, gdzie z jest dowolną spośród liter $x, y, \ldots, u$ lub inną literą nie występującą w tym ciągu, prowadzi od predykatu $P(x, y, \ldots, u, v)$ do predykatu $P(z, y, \ldots, u, v)$. Należy tylko przyjąć jedno zastrzeżenie, które wyjaśnimy na przykładzie.

Weźmy predykat:

$$P(x, y) = \mathrm{E}z\,(x < z)\,\&\,(z < y).$$

W predykacie nie możemy podstawić za x zmiennej z, gdyż zmienna x była wolna w $P(x, y)$, jeżeli zaś podstawimy za nią zmienną z, to zmienna ta będzie związana. Po podstawieniu za x zmiennej z znajdzie się ona w zasięgu kwantyfikatora $\mathrm{E}z$:

$$\mathrm{E}z[(z < z)\,\&\,(z < y)].$$

Trudność tę jest łatwo ominąć, zapisując predykat $P(x, y)$ w postaci

$$P(x, y) = \mathrm{E}t[(x < t)\,\&\,(t < y)].$$

Do predykatu tak zapisanego można wykonać podstawienie za zmienną x zmiennej z.

Pisaliśmy tu o operacjach podstawiania, a nie o operacji podstawiania, gdyż jest to nie jedna operacja a wiele operacji podstawienia za dowolną zmienną w predykacie innej zmiennej (z zastrzeżeniem by zmienne wolne nie przeszły na zmienną związaną).

Streszczenie

Stosując odpowiednią ilość razy kwantyfikatory do dowolnej funkcji zdaniowej, otrzymamy z niej zdanie. Jeżeli zaś kwantyfikatory nie wiążą wszystkich zmiennych występujących w danej funkcji zdaniowej, to przekształcają ją w pewną inną funkcję zdaniową. Rola kwantyfikatorów w takim przypadku przypomina rolę spójników zdaniowych, które z jednych zdań bądź funkcji zdaniowych pozwalały tworzyć nowe zdania lub funkcje zdaniowe.

Podobnie podstawienie za jakąś zmienną innej zmiennej tworzy z funkcji zdaniowej inną funkcją zdaniową. Podstawienie jest wykonalne tylko wtedy, gdy żadna zmienna wolna nie stanie się po podstawieniu zmienną związaną.

Zadania

1. Które z wyrażeń są zdaniami, a które tylko funkcjami zdaniowymi:

$$P(x, y, z);\quad \mathrm{A}x\,P(x, y, z);\quad \mathrm{E}x\,\mathrm{A}x\,P(x, y, z);\quad \mathrm{A}z\,\mathrm{E}y\,\mathrm{A}x\,P(x, y, z),$$

gdzie $P(x, y, z)$ jest pewnym predykatem zmiennych x, y, z. Wyróżnić zmienne wolne w tych funkcjach zdaniowych. Porównać z zadaniem 2 z poprzedniego paragrafu.

2. Podać miejsca geometryczne następujących funkcji zdaniowych:
a) $\mathrm{E}y\,(y < x)$; b) $\mathrm{A}x\,\mathrm{E}y\,(y < x)$;
c) $\mathrm{A}y\,(y < x)$; d) $\mathrm{A}x[(x < 1) \vee (x \geqslant 1)]$.

3. Znaleźć wykres predykatu:

$$P(x, y) = (x < y) \vee (x^2+y^2 = 1)$$

oraz predykatu $P(y, y)$.

4. Co otrzymujemy podstawiając do całki $\int_0^x f(t, x)\,dt$, za zmienną x zmiennej t, a potem za zmienną t zmiennej x. Porównać wynik z zastrzeżeniem podanym w tym paragrafie.

§ 19. ELEMENTARNY RACHUNEK KWANTYFIKATORÓW

W poprzednich paragrafach poznaliśmy przykłady formuł poprawnie zbudowanych w rachunku kwantyfikatorów. W ostatnim paragrafie zobaczyliśmy jakie operacje można na tych formułach dokonywać, aby w wy-

niku ich otrzymać znowu formuły poprawnie zbudowane. Obecny paragraf jest jakby zebraniem i uściśleniem tych wiadomości i zawiera określenie formuł poprawnie zbudowanych, tzw. elementarnego rachunku kwantyfikatorów. Weźmy jakiś zbiór predykatów

(1) $$P(x), \quad \ldots, \quad Q(x, y), \quad \ldots, \quad R(x, y, z), \quad \ldots,$$

jedno-, dwu-, trój- i więcej argumentowych. Predykaty te nie są konkretnymi ustalonymi funkcjami zdaniowymi, lecz tzw. *zmiennymi predykatywnymi*, za które możemy podstawiać konkretne funkcje zdaniowe. Podobnie w rachunku zdań, litery p, q, r nie były konkretnymi zdaniami, lecz zmiennymi zdaniowymi, za które mogliśmy podstawiać konkretne funkcje zdaniowe.

Określimy teraz w sposób indukcyjny pojęcie *formuły poprawnie zbudowanej* wychodząc od formuł najprostszych i przechodząc do formuł złożonych.

Definicja składa się z czterech punktów:

1. Zmienne predykatywne (1) są formułami poprawnie zbudowanymi.

2. Jeżeli $\Phi = \Phi(x, y, \ldots)$ oraz $\Psi = \Psi(x, y, \ldots)$ są formułami poprawnie zbudowanymi, to

$$\Phi \vee \Psi, \quad \Phi \,\&\, \Psi, \quad \Phi \Rightarrow \Psi, \quad \Phi \equiv \Psi \quad \text{oraz} \quad \sim \Phi$$

są formułami poprawnie zbudowanymi. Przyjmujemy przy tym podobne reguły posługiwania się nawiasami, co dla rachunku zdań (zob. § 7), tzn. takie, by formuła mogła być jednoznacznie odczytana.

3. Jeżeli $\Phi(x, y, \ldots)$ jest formułą poprawnie zbudowaną, to

$$\Psi(y, z, \ldots) = \mathrm{A}x\Phi(x, y, z, \ldots) \quad \text{oraz} \quad \Theta(y, z, \ldots) = \mathrm{E}x\Phi(x, y, z, \ldots)$$

są formułami poprawnie zbudowanymi. Zmiennymi wolnymi w formułach Ψ oraz Θ są te zmienne, które były wolne w formule $\Phi(x, y, z, \ldots)$ z wyłączeniem zmiennej x, która została związana kwantyfikatorem.

4. Jeżeli $\Phi(x, y, z, \ldots)$ jest formułą poprawnie zbudowaną, to formuła

$$\Psi(t, y, z, \ldots) = \Phi(t, y, z, \ldots),$$

uzyskana z formuły Φ przez podstawienie za zmienną wolną x zmiennej t wszędzie tam, gdzie zmienna ta występuje w formule Φ, jest formułą

poprawnie zbudowaną. Przyjmujemy tutaj jedno zastrzeżenie dotyczące wykonalności operacji podstawienia: przy operacji podstawiania zmienna t nie może się stać zmienną związaną, na żadnym miejscu w którym występuje. Kwestię tę omówiliśmy zresztą na końcu poprzedniego paragrafu.

Formuły poprawne, nazywać będziemy krótko *formułami* lub *formułami elementarnymi*. W formułach tych występują dwa rodzaje zmiennych: zmienne indywidualne $x, y, z, \ldots$ przebiegające jakiś zbiór N przedmiotów oraz zmienne predykatywne $P, Q, R, \ldots$ przebiegające predykaty jedno-, dwu-, trójargumentowe itd.

Jeżeli za zmienne predykatywne przyjmiemy pewne konkretne funkcje zdaniowe określone na zbiorze N, to formuła $\Phi(x, y, \ldots)$ będzie pewną konkretną, określoną na zbiorze N, funkcją zdaniową tych zmiennych, które są wolne w formule.

Na przykład, jeżeli w formule

(2) $$\Phi(x) = \mathrm{E}y[P(x) \vee Q(x, y)]$$

będziemy interpretowali zmienną predykatywną P jako funkcję zdaniową $x = y$ na zbiorze liczb naturalnych, zmienną zaś predykatywną $Q(x, y)$ — jako funkcję $x > y$, to formuła $\Phi(x)$ oznacza funkcję zdaniową

$$\mathrm{E}y[(x = y) \vee (x > y)]$$

jednej zmiennej x. Jest to funkcja zdaniowa prawdziwa w zbiorze liczb naturalnych (tzn. spełniona dla każdej liczby naturalnej x, por. § 15). O formule (2) mówimy, że jest spełniona przez funkcje zdaniowe $x = y$ oraz $x > y$.

Zmienne predykatywne $P(x)$ oraz $Q(x, y)$ możemy intrepretować jednak jako dowolne funkcje zdaniowe. Interpretując P jako $x \neq x$, Q zaś tak jak poprzednio, otrzymamy jako interpretację formuły (2) funkcję zdaniową

$$\mathrm{E}y[(x \neq x) \vee (x > y)]$$

spełnialną, ale już nie prawdziwą. Na przykład liczba $x = 1$, spełnia tę funkcję, gdyż $\mathrm{E}y[(1 \neq 1) \vee (1 > y)]$ jest zdaniem prawdziwym, liczba zaś $x = 0$ nie spełnia jej, gdyż nie istnieje taka liczba y, dla której $(0 \neq 0) \vee (0 > y)$.

O *formule* (2) mówimy, że jest *spełnialna* w zbiorze liczb naturalnych, gdy można tak interpretować zmienne predykatywne, że formuła będzie interpretowana jako funkcja zdaniowa spełnialna (a w naszym przypadku nawet prawdziwa). Formuła (2) nie jest jednak prawdziwa w zbiorze liczb naturalnych, gdyż przy pewnej interpretacji jej zmiennych predykatywnych nie otrzymujemy funkcji zdaniowej prawdziwej.

Formułę nazywamy *prawdziwą* w zbiorze N, jeżeli przy każdej interpretacji zmiennych predykatywnych, jako funkcji zdaniowych w zbiorze N, otrzymujemy funkcję zdaniową prawdziwą przy dowolnych wartościach zmiennych. Przykładem formuły prawdziwej jest na przykład formuła

$$\mathrm{A}x[Q(x, y) \vee \sim Q(x, y)]$$

Jest ona spełniona przy każdej interpretacji zmiennej predykatywnej Q jako funkcji zdaniowej dwuargumentowej w zbiorze N.

Streszczenie

Podsumowaliśmy rozważania poprzednich paragrafów i podaliśmy określenie formuły poprawnie zbudowanej ze zmiennych predykatywnych i symboli funktorów $\vee$, $\&$, $\Rightarrow$, $\equiv$, $\sim$ oraz $\mathrm{A}x$ i $\mathrm{E}x$ rachunku kwantyfikatorów. Omówiliśmy pojęcie formuły spełnionej przy jakiejś interpretacji zmiennych predykatywnych oraz pojęcie formuły prawdziwej.

Zadania

1. *Formułę* $\Phi(x)$ nazywamy *fałszywą*, jeżeli jej negacja $\sim \Phi(x)$ jest prawdziwa. Udowodnić, że jeżeli:

a) formuła $\Phi(x)$ nie jest spełnialna, to $\Phi(x)$ jest fałszywa;

b) formuła $\Phi(x)$ nie jest fałszywa, to $\Phi(x)$ jest spełnialna.

Jaki zachodzi związek między wypowiedziami a) i b)?

2. a) Udowodnić, że jeżeli $\Phi(x, y)$ jest formułą spełnialną, to $\mathrm{E}x\,\Phi(x, y)$ jest formułą spełnialną.

b) Udowodnić, że jeżeli $\mathrm{E}x\,\Phi(x, y)$ jest formułą spełnialną, to $\Phi(x, y)$ jest formułą spełnialną.

§ 20. TAUTOLOGIE RACHUNKU KWANTYFIKATORÓW

W rachunku zdań rozpatrywaliśmy schematy zdań. Szczególnie interesowały nas te schematy, które przy dowolnej interpretacji zmiennych zdaniowych dawały w wyniku zdania prawdziwe. Nazywaliśmy je tautologiami rachunku zdań. W rachunku kwantyfikatorów wprowadzonym powyżej formułami poprawnymi były już nie schematy zdań a schematy funkcji zdaniowych. Podstawiając za zmienne predykatywne występujące w formule konkretne funkcje zdaniowe, otrzymywaliśmy z danej formuły konkretną funkcję zdaniową.

Szczególnie interesują nas oczywiście formuły prawdziwe, tzn. takie schematy, które przy każdej interpretacji zmiennych predykatywnych dają funkcje zdaniowe prawdziwe. Formuły te nazwaliśmy w poprzednim paragrafie formułami prawdziwymi. Przez analogię z rachunkiem zdań nazywamy je *tautologiami rachunku kwantyfikatorów* lub dokładniej *tautologiami elementarnego rachunku kwantyfikatorów*. Teorię, której zbiorem twierdzeń jest zbiór tautologii elementarnego rachunku kwantyfikatorów nazywamy *elementarnym rachunkiem kwantyfikatorów*.

W dalszym ciągu podamy kilka prostych przykładów tautologii rachunku kwantyfikatorów.

1. *Prawa rozdzielności dla kwantyfikatorów.* Prawa te mają postać:

$$\vdash \mathrm{E}x[P(x) \vee Q(x)] \equiv \mathrm{E}xP(x) \vee \mathrm{E}xQ(x),$$

$$\vdash \mathrm{A}x[P(x) \,\&\, Q(x)] \equiv \mathrm{A}xP(x) \,\&\, \mathrm{A}xQ(x).$$

Znaku „$\vdash$" używamy jak poprzednio dla zaznaczenia, że formuła jest tautologią. Jest to zgodne z intuicyjnym rozumieniem kwantyfikatora szczegółowego.

Pierwsze prawo rozdzielności wiąże funktor kwantyfikatora szczegółowego z funktorem sumy logicznej. Drugie zaś wiąże funktor kwantyfikatora ogólnego z funktorem iloczynu logicznego. Można je odczytać następująco: *koniunkcja dwóch funkcji zdaniowych jest prawdziwa wtedy i tylko wtedy, gdy każda z tych funkcji zdaniowych jest prawdziwa.* Jest to zgodne z intuicyjnym rozumieniem kwantyfikatora ogólnego.

2. *Prawa de Morgana dla kwantyfikatorów.* Prawa te przypominają znane nam prawa de Morgana dla rachunku zdań. Są one zresztą ich

uogólnieniem na przypadek, kiedy operujemy nieskończonymi alternatywami i koniunkcjami. Prawa te zapiszemy w postaci:

$$\vdash \sim \mathrm{E}xP(x) \equiv \mathrm{A}x \sim P(x),$$
$$\vdash \sim \mathrm{A}xP(x) \equiv \mathrm{E}x \sim P(x).$$

Pierwsze z tych praw stwierdza, że zwrot: „nie istnieje takie x, że $P(x)$" jest równoważny zwrotowi „dla każdego x nie $P(x)$". Na przykład niech $P(x)$ będzie predykatem $x > 0$ oraz niech dziedziną tego predykatu będzie zbiór liczb naturalnych (z zerem). Wtedy pierwsze prawo de Morgana przyjmie postać

$$\sim \mathrm{E}x(x > 0) \equiv \mathrm{A}x \sim (x > 0).$$

Podobnie, dla drugiego prawa de Morgana

$$\sim \mathrm{A}x(x > 0) \equiv \mathrm{E}x \sim (x > 0).$$

Prawdziwość tych praw jest również zgodna z intuicyjnym używaniem zwrotów: „istnieje" i „dla każdego". Jeżeli bowiem nieprawdą jest, że istnieje taki przedmiot, który spełnia funkcję zdaniową $P(x)$, to znaczy to samo co: prawdą jest, iż każdy przedmiot nie spełnia funkcji zdaniowej $P(x)$. I podobnie dla drugiego prawa de Morgana. Nieprawda, że każdy przedmiot x spełnia funkcję zdaniową $P(x)$, jest równoważne zwrotowi: istnieje taki przedmiot x, który nie spełnia funkcji zdaniowej $P(x)$. Mówiąc inaczej prawa de Morgana stwierdzają:

Jeżeli w jakimś zbiorze nie istnieje przedmiot mający daną własność, tzn. to samo co: każdy przedmiot należący do rozważanego zbioru nie ma danej własności.

Jeżeli nie każdy przedmiot zbioru ma daną własność, tzn. to samo co: istnieje taki przedmiot należący do zbioru, który rozważamy i który nie ma danej własności.

4. Innym przykładem tautologii jest prawo

$$\vdash \mathrm{A}xP(x) \Rightarrow P(a),$$

mówiące, że *jeżeli własność $P(x)$ przysługuje każdemu elementowi x jakiegoś zbioru, to również przysługuje ona jakiemuś konkretnemu elementowi a.* Na przykład, jeżeli dla każdej liczby naturalnej istnieje liczba od niej większa o 1, to również dla liczby 5 istnieje liczba większa od niej o 1. Innym przykładem zastosowania tej tautologii, jest stwierdzenie, że jeżeli wszyscy ludzie mają wzrost nie większy niż 2 metry, to i Pan Kowalski ma wzrost nie większy niż 2 metry. Dziedziną w tym ostatnim przykładzie są wszyscy ludzie, natomiast funkcja zdaniowa ma postać

x jest nie większy niż dwa metry.

5. Jeszcze innym przykładem tautologii, z której często robimy użytek w wielu rozumowaniach jest następująca tautologia

$$\vdash P(a) \Rightarrow \mathrm{E}xP(x)$$

Jeżeli jakiś element ma własność P, to możemy powiedzieć, że istnieje taki element x, że $P(x)$. Na przykład, jeżeli $P(a)$ oznacza, że a jest liczbą pierwszą, to możemy powiedzieć, że istnieje taka liczba x, która jest liczbą pierwszą. Albo jeżeli Pan Kowalski waży 50 kg, to możemy powiedzieć, że istnieje taki człowiek, który waży 50 kg.

Podane przykłady praw logicznych mogą się wydawać banalne, jednakże we wszelkich rozumowaniach czynimy z nich użytek w sposób świadomy lub nieświadomy.

Streszczenie

Prawa rachunku kwantyfikatorów zwane też tautologiami logicznymi stanowią uzupełnienie praw rachunku zdań i wraz z nimi pozwalają na przeprowadzenie rozumowań w każdej teorii matematycznej.

Zadania

1. Udowodnić, wstawiając za $P(x)$, $Q(x)$ oraz $R(x, y)$ odpowiednie funkcje zdaniowe, że następujące formuły nie są tautologiami:

a) $\mathrm{E}xP(x) \Rightarrow \mathrm{A}xP(x)$;

b) $Q(x) \vee \mathrm{A}yQ(y)$;

c) $\mathrm{A}x\mathrm{E}yR(x, y) \Rightarrow \mathrm{E}y\mathrm{A}xR(x, y)$.

Wskazówka. Podstawić np. $P(x) = (x \neq 0)$, $Q(x) = (x = 0)$, $R(x, y) = (x < y)$, gdzie zmienne x, y przebiegają zbiór liczb naturalnych.

2. Uzasadnić na podstawie rozumowania intuicyjnego następujące tautologie:

$$\vdash \mathrm{E}x[P(x) \,\&\, Q(x)] \Rightarrow \mathrm{E}x\,P(x) \,\&\, \mathrm{E}x\,Q(x),$$
$$\vdash \mathrm{A}x\,P(x) \vee \mathrm{A}x\,Q(x) \Rightarrow \mathrm{A}x\,[P(x) \vee Q(x)].$$

3. a) Uzasadnić za pomocą rozumowania intuicyjnego następującą tautologię:

$$\vdash \mathrm{E}x\mathrm{A}y\,R(x,y) \Rightarrow \mathrm{A}y\mathrm{E}x\,R(x,y).$$

b) Pokazać na przykładzie, że implikacja w drugą stronę nie jest tautologią.

4. Z tautologii z punktów 4 i 5 wyprowadzić, posługując się regułą odrywania i tautologią rachunku zdań: $(p \Rightarrow q) \Rightarrow ((q \Rightarrow r) \Rightarrow (p \Rightarrow r))$, następującą tautologię:

$$\vdash \mathrm{A}x\,P(x) \Rightarrow \mathrm{E}x\,P(x).$$

5. a) Uzasadnić tautologię z punktu 5 za pomocą praw de Morgana oraz tautologii z punktu 4.

b) Uzasadnić tautologię z punktu 5 za pomocą praw de Morgana oraz tautologii z punktu 5.

6. a) Uzasadnić za pomocą rozumowania intuicyjnego następujące tautologie:

$$\vdash \mathrm{A}x(Q \,\&\, P(x)) \equiv Q \,\&\, \mathrm{A}x\,P(x),$$
$$\vdash \mathrm{A}x(Q \vee P(x)) \equiv Q \vee \mathrm{A}x\,P(x),$$

gdzie funkcja zdaniowa Q nie zawiera zmiennej wolnej x.

b) Podobnie uzasadnić tautologie:

$$\vdash \mathrm{E}x(Q \vee P(x)) \equiv Q \vee \mathrm{E}x\,P(x),$$
$$\vdash \mathrm{E}x(Q \,\&\, P(x)) \equiv Q \,\&\, \mathrm{E}x\,P(x).$$

7. a) Uzasadnić za pomocą rozumowania intuicyjnego następujące tautologie:

$$\vdash (P \Rightarrow \mathrm{A}x\,Q(x)) \equiv \mathrm{A}x(P \Rightarrow Q(x)),$$
$$\vdash (\mathrm{E}x\,Q(x) \Rightarrow P) \equiv \mathrm{E}x(Q(x) \Rightarrow P),$$

gdzie funkcja zdaniowa P nie zawiera zmiennych wolnych.

b) Uzasadnić te tautologie na podstawie tautologii podanych w zadaniu 6.

8. Uzasadnić intuicyjnie prawdziwość tautologii

$$\vdash (\mathrm{E}x\,Q(x) \Rightarrow P) \equiv \mathrm{A}x\,(Q(x) \Rightarrow P).$$

9. a) Uzasadnić za pomocą rozumowania intuicyjnego następującą tautologię:

$$\vdash \mathrm{A}x\,(P(x) \Rightarrow \mathrm{A}y\,P(y)) \Rightarrow \mathrm{A}y\,P(y)$$

zwaną *prawem dołączania dużego kwantyfikatora.*

10. Uzasadnić prawdziwość implikacji:

$$\vdash \mathrm{A}x(P(x) \Rightarrow Q(x)) \Rightarrow (\mathrm{A}x\,P(x) \Rightarrow \mathrm{A}x\,Q(x)),$$
$$\vdash (\mathrm{E}x\,P(x) \Rightarrow \mathrm{E}x\,Q(x)) \Rightarrow \mathrm{E}x(P(x) \Rightarrow Q(x)).$$

11. Uzasadnić w sposób intuicyjny następujące tautologie:

$$\vdash \mathrm{A}x\mathrm{A}y\,Q(x,y) \equiv \mathrm{A}y\mathrm{A}x\,Q(x,y),$$
$$\vdash \mathrm{E}x\mathrm{E}y\,Q(x,y) \equiv \mathrm{E}y\mathrm{E}x\,Q(x,y).$$

12. Napisać zaprzeczenie formuły $Q(x) \vee \mathrm{A}y\,Q(y)$ z zadania 1b). Udowodnić, że zaprzeczenie tej formuły również nie jest tautologią.

§ 21. RACHUNEK KWANTYFIKATORÓW JAKO TEORIA DEDUKCYJNA

Omówimy tutaj reguły wnioskowania pozwalające z jednych tautologii rachunku kwantyfikatorów uzyskiwać inne.

1. *Podstawianie funkcji zdaniowych do tautologii rachunku zdań.* Jeżeli formuła $\Phi(p, q, \ldots, t)$ jest tautologią rachunku zdań, to po wstawieniu za zmienne zdaniowe $p, q, \ldots, t$ dowolnych funkcji zdaniowych otrzymamy tautologię rachunku kwantyfikatorów:

$$\vdash \Phi\big(A(x), B(x), \ldots, C(x)\big).$$

Na przykład wyrażenia:

$$\vdash A(x) \vee \sim A(x),$$
$$\vdash A(x) \Rightarrow A(x),$$
$$\vdash \sim\sim A(x) \equiv A(x),$$
$$\vdash A(x) \vee B(x) \equiv B(x) \vee A(x)$$

są tautologiami rachunku kwantyfikatorów.

2. *Reguła odrywania.* Jeżeli $\vdash A(x)$ oraz implikacja

$$\vdash A(x) \Rightarrow B(x)$$

są tautologiami rachunku kwantyfikatorów, to również wyrażenie $\vdash B(x)$ jest tautologią rachunku kwantyfikatorów.

3. *Reguła podstawiania.* Jeżeli w tautologii $\vdash A(x)$, podstawienie za zmienną x zmiennej z jest wykonalne (tzn. zmienna wolna x nie przechodzi na zmienną związaną), to tautologią będzie również wyrażenie $\vdash A(z)$.

4. *Reguła uogólnienia.* Jeżeli tautologią jest $\vdash A(x)$ to tautologią jest również wyrażenie

$$\mathrm{A}x\,A(x)$$

Prawdziwość reguł 2-4 jest oczywista.

Reguły 1-4 są oczywiście regułami prowadzącymi od zdań intuicyjnie prawdziwych do zdań intuicyjnie prawdziwych.

Przyjmuje się je zazwyczaj jako reguły wnioskowania w rachunku kwantyfikatorów.

Jako aksjomaty rachunku kwantyfikatorów można przyjąć na przykład następujący zespół aksjomatów:

F_1. Wszystkie tautologie rachunku zdań.

F_2. Tautologie postaci:

$$\vdash \mathrm{A}x\big(A \Rightarrow B(x)\big) \Rightarrow \big(A \Rightarrow \mathrm{A}xB(x)\big),$$

$$\vdash \mathrm{A}x\big(B(x) \Rightarrow A\big) \Rightarrow \big(\mathrm{E}xB(x) \Rightarrow A\big),$$

gdzie funkcja zdaniowa A nie zawiera zmiennej wolnej x.

F_3. $\vdash \mathrm{A}xB(x) \Rightarrow B(y)$,

F_4. $\vdash \mathrm{E}x\big(B(x) \vee \sim B(x)\big)$.

Ciekawą rolę gra aksjomat F_4. Wymaga on by zakres zmienności kwantyfikatora Ex, a więc Ax był niepusty.

Zbiór formuł, które można otrzymać z aksjomatów F_1-F_4 za pomocą reguł 1-4, jest zbiorem twierdzeń elementarnego rachunku kwantyfikatorów. Każde twierdzenie jest tautologią rachunku kwantyfikatorów, gdyż aksjomaty są tautologiami, reguły zaś wnioskowania prowadzą od formuł prawdziwych (czyli tautologii), do formuł prawdziwych — tautologii. Co więcej wszystkie znane tautologie dają się wyprowadzić z aksjomatów F_1-F_4 za pomocą reguł 1-4.

Rachunek kwantyfikatorów można ujmować aksjomatycznie na wiele sposobów, podając (por. zadanie do tego paragrafu) różne reguły dedukcji i różne aksjomaty.

Podany przez nas układ aksjomatów jest nieskończony, gdyż w F_1 jest nieskończona liczba aksjomatów. Możliwe są oczywiście inne układy złożone ze skończonej liczby aksjomatów.

Rozdział ten zakończymy pewnymi uwagami końcowymi omawiającymi różnice między rachunkiem zdań a rachunkiem kwantyfikatorów.

W rachunku zdań sens spójników zdaniowych określiliśmy w sposób ścisły metodą zero-jedynkową. Dzięki temu mieliśmy prostą i matematycznie ścisłą metodę sprawdzania czy formuła jest tautologią, tj. czy jest zawsze prawdziwa. Podstawialiśmy za zmienne zdaniowe wszelkie możliwe kombinacje zer i jedynek i obliczaliśmy wartość logiczną całego zdania dla każdego podstawienia. O ile dla każdego podstawienia wartość ta była równa jedności, wyrażenie było tautologią.

Dla rachunku kwantyfikatorów podobna metoda nie istnieje gdyż zbiory przedmiotów na których określone są funkcje zdaniowe są nieskończone. Nie możemy więc podstawić wszelkich możliwych nazw przedmiotów za zmienne w funkcjach zdaniowych i sprawdzić dla każdego podstawienia, czy badana funkcja zdaniowa jest spełniona. W przypadku skończonych zbiorów przedmiotów mielibyśmy w istocie do czynienia nie z rachunkiem kwantyfikatorów a z rachunkiem zdań, a jak pamiętamy, pojęcie kwantyfikatora jest wtedy zbędne, zastępuje je bowiem zwykła suma i iloczyn logiczny. Zbiory przedmiotów rozważanych w matematyce są na ogół nieskończone, więc zastosowanie metody podobnej do metody zero-jedynkowej jest dla rachunku kwantyfikatorów niemożliwe.

Choć pojęcie tautologii jest intuicyjnie zrozumiałe, to jedynie ścisłą i poprawną metodą określania tautologii rachunku kwantyfikatorów jest metoda aksjomatyczna.

Czytelnika, który byłby zainteresowany aksjomatycznym ujęciem rachunku kwantyfikatorów i nie chciał poprzestać na podanym przez nas szkicu odsyłamy do specjalnych podręczników logiki, podanych na końcu książki.

Streszczenie

Podaliśmy reguły wnioskowania (reguły 1-4) i aksjomaty (F_1-F_4) elementarnego rachunku kwantyfikatorów. Twierdzeniami są te formuły, które dają się wyprowadzić z aksjomatów za pomocą reguł dedukcji. Wszystkie twierdzenia tego rachunku są tautologiami.

Zadania

1. Uzasadnić intuicyjnie, że reguły 1-4 prowadzą od zdań prawdziwych do zdań prawdziwych.

2. Wyprowadzić z reguł 1-4 i aksjomatów następujące reguły pochodne:

a) regułę dołączania dużego kwantyfikatora prowadzącego od tautologii $\vdash A \Rightarrow B(x)$ do tautologii $\vdash A \Rightarrow \mathrm{A}xB(x)$;

b) regułę opuszczania dużego kwantyfikatora prowadzącą od tautologii $\vdash A \Rightarrow \mathrm{A}xB(x)$ do tautologii $\vdash A \Rightarrow B(x)$;

c) regułę dołączania małego kwantyfikatora prowadzącą od tautologii $\vdash B(x) \Rightarrow A$ do tautologii $\vdash \mathrm{E}xB(x) \Rightarrow A$;

d) regułę opuszczania małego kwantyfikatora, prowadzącą od tautologii $\vdash \mathrm{E}xB(x) \Rightarrow A$ do tautologii $\vdash B(x) \Rightarrow A$.

Uzasadnić te reguły w sposób intuicyjny.

3. Uzasadnić, że aksjomaty F_2 i F_3 można by zastąpić regułami a), b), c), d) z zadania 2.

4. Uzasadnić, że oba aksjomaty F_2 można zastąpić prawem de Morgana:

$$\vdash \sim \mathrm{A}x(A(x)) \equiv \mathrm{E}x(\sim A(x)).$$

Rozdział 4

TEORIE ELEMENTARNE

W rozdziale tym zajmiemy się sformalizowanymi teoriami matematycznymi. Omówimy mianowicie klasę teorii zwanych *teoriami elementarnymi.*

Sformalizowane teorie matematyczne posługują się językiem rachunku zdań oraz rachunku kwantyfikatorów, wzbogaconym tylko o pojęcie funkcji (często używamy zresztą w tym przypadku nie terminu funkcja a terminu działanie). W formułach sformalizowanych teorii matematycznych występują pewne ustalone predykaty, których własności są opisane za pomocą aksjomatów, które te predykaty muszą spełniać. Aksjomaty te — w odróżnieniu od aksjomatów logicznych — nazywają się *aksjomatami specyficznymi* teorii. W odróżnieniu od aksjomatów logicznych wspólnych dla wszystkich sformalizowanych teorii matematycznych, aksjomaty specyficzne teorii są różne; dla różnych teorii opisują różne własności pojęć pierwotnych.

Teorie elementarne są to sformalizowane teorie matematyczne o możliwie najprostszym języku. W formułach tych teorii nie występują zwroty: „dla każdego predykatu ...", czy też „istnieje taki predykat ...", choć mogą występować zwroty: „dla każdego przedmiotu ..." czy też „istnieje taki przedmiot ..." Mówiąc ściślej formuły poprawnie zbudowane teorii elementarnych — tzw. *formuły elementarne* zawierają symbole kwantyfikatorów Ax oraz Ex, których zakres zmienności jest zbiorem przedmiotów. Nie mogą natomiast występować kwantyfikatory, których zakresem zmienności jest jakiś zbiór predykatów. Wyrażenie: „dla każdego predykatu $P(x)$ zachodzi $P(x)$" nie da się zapisać w sformalizowanym języku teorii elementarnych. Można w tym języku zapisać jedynie wyrażenie: „dla każdego przedmiotu x, zachodzi $P(x)$" (gdzie $P(x)$ jest jakimś predykatem).

Teorie elementarne stanowią najprostszy typ sformalizowanych teorii matematycznych. Wprawdzie język ich nie jest dostatecznie bogaty by można było w nim wyrazić całą matematykę, jednak nadają się świetnie do ilustracji zagadnień związanych z formalizacją matematyki.

W paragrafie 22 zajmiemy się działaniami i ich przedstawieniem za pomocą termów. W paragrafie 23 omówimy język teorii elementarnych. W paragrafie 24 zastanowimy się co ten język opisuje. Kwestią reguł dedukcji aksjomatów i dowodzenia twierdzeń zajmiemy się w paragrafie 25. W paragrafie 26 omówimy pojęcie modelu teorii elementarnej oraz zagadnienie prawdziwości twierdzeń teorii.

W ostatnim paragrafie 27 zajmiemy się, tylko szkicowo, zagadnieniami sformalizowanych teorii matematyki, które nie są już teoriami elementarnymi, tzw. zagadnieniami teorii nieelementarnych oraz szkicowym omówieniem teorii sformalizowanych o regułach dowodzenia nie związanych z prawami logiki.

Czytelnikowi, który nie jest zainteresowany studiowaniem teorii elementarnych, a chciałby zdobyć podstawowe wiadomości o formalizacji matematyki zalecamy pominięcie tego rozdziału i przeczytanie tylko ostatniego jego paragrafu (§ 27), oraz ewentualny powrót do tego rozdziału w trakcie czytania dalszych części książki.

Sformalizowane teorie matematyczne mają bogatą literaturę. Czytelnika, którego interesują te zagadnienia odsyłamy do następującej literatury: Grzegorczyk [1961], Rasiowa i Sikorski, Kleene [1952], Mostowski.

§ 22. TERMY I DZIAŁANIA

Rozważania nad językami matematycznych teorii sformalizowanych rozpoczniemy od wyjaśnienia pojęcia termów. W matematyce rozważa się różne działania np. dodawanie, mnożenie, podnoszenie do kwadratu i inne. Wyrażenia zbudowane w pewien specjalny sposób ze zmiennych i symboli oznaczających te działania nazywamy *termami*. Na przykład wyrażenia $(x+y)\cdot z$ oraz $(x\cdot y)+z$ są termami zbudowanymi ze zmiennych x, y, z oraz symboli „+" i „·" oznaczających działania dodawania i mnożenia.

Zanim przejdziemy do dalszego wyjaśniania pojęcia termu, rozważymy pewne kwestie dotyczące zapisu wyrażeń matematycznych.

Często zamiast $x+y$ przyjmuje się zapis $+(x, y)$, względnie zamiast $x\cdot y$ — zapis $\cdot(x, y)$ stawiając symbol działania na pierwszym miejscu, a po nim argumenty, ujęte we wspólny nawias. Zgodnie z tą konwencją term $(x+y)\cdot z$ możemy zapisać $\cdot\big(+(x, y), z\big)$ term $(x\cdot y)+z$ zaś możemy zapisać jako $+\big(\cdot(x, y)\, z\big)$.

Taki sposób pisania budzić może pewne opory w przypadku działań dwuargumentowych, gdzie przyzwyczajeni jesteśmy do pisania znaku działania między argumentami, jednak bez oporów posługujemy się nim w przypadku działań o innej liczbie argumentów np. piszemy n.w.d. $(x_1, \ldots, x_n)$ i rozumiemy, że symbol n.w.d. (traktowany jako jeden znak) jest symbolem działania (obliczania największego wspólnego dzielnika), argumentami zaś działania są zmienne $x_1, \ldots, x_n$.

Działania jednoargumentowe przyjęto pisać na obydwa sposoby, np. działanie obliczania elementu przeciwnego do x zapisujemy jako $-x$ (w nowej konwencji, poprawniej $-(x)$), ale działanie podnoszenia do kwadratu zapisujemy jako x^2, choć czytelnikowi zapewne nie obcy jest zapis $\mathrm{sqrt}\, x$, dostosowany do nowej konwencji, w której symbol sqrt oznacza znak działania podnoszenia do kwadratu.

Warto przy okazji zauważyć, że można rozważać również działania zeroargumentowe. Za działanie zeroargumentowe możemy uważać każdą funkcję, której wartość równa się ustalonemu elementowi (funkcje o wartości stałej).

Powróćmy teraz do dalszych rozważań nad termami.

Powiedzieliśmy, że termy są pewnymi wyrażeniami, zbudowanymi ze znaków działań i zmiennych, nie precyzując ściśle jakie to mają być wyrażenia, np. nie wiemy czy wyrażenie $+(x, y)\cdot(x, y)$ złożone z symboli działań „+" oraz „·" zmiennych x i y mamy uważać za term, czy też nie. Podamy więc teraz definicję termu.

Niech litery $x_1, x_2, x_3, \ldots$ będą zmiennymi. Nie będziemy na razie precyzować jakie elementy można za te zmienne podstawiać.

Weźmy pewien zbiór D działań złożony ze skończonej lub nieskończonej liczby działań zeroargumentowych: $c_1, c_2, \ldots$, jednoargumentowych: $g_1(x_1)$, $g_2(x_2), \ldots$, dwuargumentowych: $k_1(x_1, x_2)$, $k_2(x_1, x_2), \ldots$, trójargumentowych: $t_1(x_1, x_2, x_3)$, $t_2(x_1, x_2, x_3), \ldots$, i tak dalej.

Definiując termy względem działań ze zbioru D, tak jak poniżej musimy określić jednocześnie co to są zmienne termu.

Termy zdefiniujemy indukcyjnie:

1. Termami są symbole $x_1, x_2, x_3, \ldots$ (symbole oznaczające zmienne) oraz symbole $c_1, c_2, \ldots$ (symbole działań zeroargumentowych). Zmienną termu x_i jest zmienna x_i. Dla termu c_j zbiór zmiennych jest pusty — nie zawiera żadnych elementów.

W powyższy sposób określiliśmy najprostsze termy. Termy złożone określimy następująco:

2. Jeżeli $f_1(x_{\alpha_1}, \ldots, x_{\alpha_s}), \ldots, f_n(x_{\beta_1}, \ldots, x_{\beta_r})$ są termami zmiennych podanych w nawiasie, $f(x_1, \ldots, x_n)$ zaś działaniem n-argumentowym ze zbioru D, $n > 0$, to wyrażenie $f\big(f_1(x_{\alpha_1}, \ldots, x_{\alpha_s}), \ldots, f_n(x_{\beta_1}, \ldots, x_{\beta_r})\big)$ jest termem. Zbiór zmiennych tego termu składa się ze wszystkich zmiennych występujących w ciągu $x_{\alpha_1}, \ldots, x_{\alpha_s}, \ldots, x_{\beta_1}, \ldots, x_{\beta_r}$.

Łatwo zauważyć, że przy tej definicji symbol $f(x_1, \ldots, x_n)$ którym oznaczaliśmy działanie n-argumentowe, jest termem zmiennych $x_1, \ldots, x_n$.

Nie wszystkie jednak napisy są termami, np. wspomniane już wyrażenie $+(x, y)\cdot(x, y)$ nie jest termem.

Definicja termów jest czysto formalna, termy są niczym innym jak pewnymi wyrażeniami złożonymi z pewnych znaków. Warto jednak zastanowić się, skąd powstała taka definicja termu i dlaczego jest użyteczna.

W tym celu rozważamy dokładnie co to jest działanie. Weźmy zbiór X jakichś elementów i funkcję $f(x_1, \ldots, x_n)$, która każdemu układowi n elementów $x_1, \ldots, x_n$ ze zbioru X przyporządkowuje pewien element y ze zbioru X. Element Y nazywamy wartością tej funkcji dla argumentów $x_1, \ldots, x_n$ i oznaczamy przez $f(x_1, \ldots, x_n)$. Funkcję taką nazywamy *działaniem n-argumentowym* określonym w zbiorze X.

Działaniem zeroargumentowym nazywamy ustalony element zbioru X. Działań takich określonych w zbiorze może być wiele, można je uważać za funkcje, których wartości są stałe. Każdy element należący do X może więc wyznaczać działanie zeroargumentowe — funkcję, której wartość jest stała i równa się temu elementowi.

Należy również zauważyć, że funkcja tożsamościowa przyporządkowująca każdemu elementowi x ten element tzn. x jest również działaniem.

Działania można ze sobą składać i otrzymywać nowe działania. Jeżeli f jest działaniem n-argumentowym, $f_1, \ldots, f_n$ zaś są działaniami odpowiednio

s-, …, r-argumentowymi to złożeniem działania f z działaniami $f_1, \ldots, f_n$ jest funkcja, która każdemu układowi argumentów $x_{\alpha_1}, \ldots, x_{\alpha_s}, \ldots, x_{\beta_1}, \ldots, x_{\beta_r}$ przyporządkowuje element równy.

$$f\big(f_1(x_{\alpha_1}, \ldots, x_{\alpha_s}), \quad \ldots, \quad f_n(x_{\beta_1}, \ldots, x_{\beta_r})\big).$$

Złożenie działań jest również działaniem.

Zajmijmy się teraz kwestią interpretacji termów. Termy są pewnymi napisami wyznaczającymi schemat kolejnego składania działań. Jeżeli zmienne $x_1, x_2, \ldots$ będą przyjmować wartości $x_1, x_2, \ldots$, symbole zaś ze zbioru D będziemy uważać za pewne funkcje określone w zbiorze X, odpowiednio zero-, jedno-, dwu-, trój- lub więcej argumentowe o wartościach z X, czyli za działania w zbiorze X, to term $f(x_1, \ldots, x_n)$ wyznaczy pewne działanie. Działanie to przyporządkowuje elementom $x_1, \ldots, x_n$ ze zbioru X element $y = f(x_1, \ldots, x_n)$ z tego samego zbioru. Znając dany term i wiedząc jakie działania w nim występują, będziemy wiedzieli, w jaki sposób zostało otrzymane działanie opisane tym termem, jako złożenie działań ze zbioru D.

Termy służą więc do opisu działań; stąd ich podstawowa rola w języku służącym do opisu jakiejkolwiek teorii matematycznej. W następnym paragrafie podamy w jaki sposób termy występują w językach matematycznych teorii sformalizowanych, a ściślej mówiąc, w językach teorii elementarnych.

Streszczenie

Podaliśmy definicję termów, definicję działań określonych na jakimś zbiorze oraz zastanowiliśmy się jak termy opisują działania.

Zadania

1. Określmy długość termu następująco:

1) Długość termów x_i oraz c_i $(i = 1, 2, 3, \ldots)$ jest równa 1.

2) Długość termu $f(f_1(\ldots), \ldots, f_n(\ldots))$ dla f ze zbioru D jest równa 1 plus suma długości termów $f_1, \ldots, f_n$.

Udowodnić, że długość termu równa się ilości liter w nim występujących, jeżeli nie bierzemy pod uwagę nawiasów, przecinków, kropek oraz indeksów.

2. Wypisać wszystkie termy o długości równej 5 złożone z działań: g jednoargumentowego oraz h dwuargumentowego, w których występują: a) zmienne x_1, x_2, b) zmienne $x_1, \ldots, x_n$.

3. Udowodnić, że jeżeli $f(x_1, \ldots, x_n)$ jest termem o zmiennych $x_1, \ldots, x_n$ oraz $f_1(x_1, \ldots, x_k), \ldots, f_n(x_1, \ldots, x_k)$ są termami, to wyrażenie $f(f_1(x_1, \ldots, x_k), \ldots, f_n(x_1, \ldots, x_k))$ powstałe przez podstawienie za zmienne $x_1, \ldots, x_n$ w termie f termów $f_1, \ldots, f_n$ jest także termem.

4. a) Podać definicję i przykłady termów w symbolice beznawiasowej, gdzie działanie $f(x_1, \ldots, x_n)$ zapisujemy bez użycia nawiasów pisząc $fx_1, \ldots, x_n$.

b) Udowodnić, że jeżeli dla każdego działania występującego w beznawiasowym zapisie termu znana jest ilość argumentów, to term może być jednoznacznie odczytany jako złożenie wykonywanych w nim działań.

5. Udowodnić, że złożenie działania n-argumentowego z działaniami odpowiednio s_1-argumentowym, ..., s_n-argumentowym jest działaniem $(s_1 + \ldots + s_n)$-argumentowym.

§ 23. JĘZYK TEORII ELEMENTARNYCH

W paragrafie 19 określiliśmy język elementarnego rachunku kwantyfikatorów. Pokrótce przypomnimy te rozważania.

Wychodziliśmy tam z pewnego zbioru elementów zwanych zmiennymi (ściślej zmiennymi indywidualnymi)

(1) $$x_1, \quad x_2, \quad \ldots$$

oraz z pewnych zmiennych predykatywnych

(2) $$P(x_1), \quad \ldots, \quad Q(x_1, x_2), \quad \ldots, \quad R(x_1, x_2, x_3), \quad \ldots$$

dla oznaczania funkcji zdaniowych jedno-, dwu-, trzy- i więcej argumentowych.

Formuły poprawnie zbudowane określiliśmy indukcyjnie.

1. Wyrażenia (2) są formułami poprawnie zbudowanymi. Formuły takie nazywamy *formułami atomowymi*.

2. Łącząc formuły poprawnie zbudowane spójnikami logicznymi $\vee$, $\&$, $\Rightarrow$, $\equiv$, $\sim$, alternatywy, koniunkcji, implikacji, równoważności i negacji, otrzymujemy również formuły poprawnie zbudowane.

3. Dopisując przed formułą poprawnie zbudowaną kwantyfikator szczegółowy lub ogólny, względem zmiennej x_i $(i = 1, 2, 3, \ldots)$ otrzymujemy również formułę poprawnie zbudowaną.

4. Podstawiając za zmienną x_i zmienną x_j w formule poprawnie zbudowanej, otrzymujemy formułę poprawnie zbudowaną.

Dla każdej formuły wyróżniliśmy ponadto które zmienne są wolne a które związane.

Język teorii elementarnych różni się tylko nieznacznie od języka elementarnego rachunku kwantyfikatorów. Oprócz symboli (1) zmiennych indywidualnych i symboli (2) zmiennych predykatywnych występuje pewien zbiór D symboli

(3) $$c_1, c_2, \ldots, g_1(x_1), g_2(x_2), \ldots, k_1(x_1, x_2), \ldots, t_1(x_1, x_2, x_3), \ldots$$

działań zero-, jedno-, dwu-, trzy- itd. argumentowych. Porównaj § 22 tego rozdziału.

Symbole operacji logicznych $\vee$, $\&$, $\Rightarrow$, $\equiv$, $\sim$, Ex_i oraz $Ax_i (i = 1,2, \ldots)$ pozostają te same co poprzednio.

Definicja formuł poprawnie zbudowanych jest taka sama jak poprzednio, z tą różnicą, że operacja podstawiania jest inaczej określona. Za zmienne wolne w formule możemy podstawiać nie tylko inne zmienne, lecz również termy, przyjmując zastrzeżenie, by po podstawieniu żadna zmienna termu nie stała się zmienną związaną.

Symbolicznie możemy tak sformułowaną regułę podstawiania zapisać następująco

4′. Jeżeli $\Phi(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n)$ jest formułą poprawnie zbudowaną o zmiennych wolnych $x_1, \ldots, x_n$, $f = f(x_{a_1}, \ldots, x_{a_s})$ zaś jest termem, to wyrażenie

(4) $$\Phi(x_1, \ldots, f(x_{a_1}, \ldots, x_{a_s}), \ldots, x_n)$$

jest formułą poprawnie zbudowaną powstałą przez wstawienie za zmienną x_i termu f wszędzie tam, gdzie zmienna ta występuje, z tym zastrzeżeniem, że żadna ze zmiennych $x_{a_1}, \ldots, x_{a_s}$ nie stanie się zmienną związaną. Zmiennymi formuły są zmienne $x_1, \ldots, x_{i-1}, x_{a_1}, \ldots, x_{a_s}, x_{i+1}, \ldots, x_n$. Oczywiście zmienne te mogą się powtarzać.

Reguła 4′ dopuszcza więcej podstawień niż reguła 4 z paragrafu 19. Reguła 4 pozwalała podstawiać za zmienne tylko zmienne, reguła 4′ zaś pozwala za zmienne wstawiać termy.

Powoduje to, że zbiór formuł poprawnych teorii elementarnej jest bogatszy od zbioru formuł poprawnych elementarnego języka rachunku

kwantyfikatorów. W formułach poprawnie zbudowanych mogą występować symbole działań oraz termy.

Poniżej podamy przykład wyjaśniający pojęcie formuły poprawnie zbudowanej. W alfabecie użytym w naszym przykładzie występuje jeden symbol predykatu dwuargumentowego Q oraz jeden symbol działania dwuargumentowego f. Dla większej przejrzystości zapisu będziemy predykat Q oznaczali symbolem „=", działanie f zaś symbolem „+" oraz będziemy pisali $x_1 = x_2$ zamiast $Q(x_1, x_2)$, oraz $x_1 + x_2$ zamiast $f(x_1, x_2)$.

Wyrażenia:

$$x_1 = x_2, \quad x_1 = x_1, \quad x_1 = (x_1 + x_2) + x_3$$

będą formułami poprawnie zbudowanymi. Pierwsza formuła ma dwie, druga jedną, trzecia trzy zmienne wolne. Dwa ostatnie wyrażenia powstają z pierwszego przez podstawienie: w pierwszym wypadku za zmienną x_2 termu x_1 (z definicji, pojedyncza zmienna też jest termem), drugie przez podstawienie za zmienną x_2 termu $(x_1 + x_2) + x_3$.

Podobnie formułami poprawnie zbudowanymi będą wyrażenia:

$$\sim \big((x_1 = x_2) \Rightarrow (x_1 = x_1)\big),$$
$$(x_1 = x_1) \vee (x_1 = (x_1 + x_2) + x_3).$$

Łatwo również podać przykłady formuł zawierających kwantyfikatory. Na przykład formułą o zmiennej wolnej x_1 będzie wyrażenie

$$\mathrm{A}x_3\, \mathrm{E}x_2 (x_1 = (x_1 + x_2) + x_3). \tag{5}$$

W formule (5) możemy za zmienną wolną x_1 podstawiać dowolny term zbudowany zé znaku „+" oraz zmiennych $x_1, x_4, x_5, \ldots$, i otrzymamy formułę poprawnie zbudowaną, np. wstawiając za x_1 term $(x_1 + x_4) + x_5$, otrzymamy formułę

$$\mathrm{A}x_3\, \mathrm{E}x_2 \big((x_1 + x_4) + x_5 = \big(\big((x_1 + x_4) + x_5\big) + x_2\big) + x_3\big)$$

o zmiennych wolnych x_1, x_4, x_5. Zmienne x_2 oraz x_3 będą związane.

W formule (5) podstawienia za zmienną wolną x_1 termu $x_1 + x_2$ wykonać nie możemy. Po tym podstawieniu otrzymamy formułę

$$\mathrm{A}x_3\, \mathrm{E}x_2 \big(x_1 + x_2 = \big((x_1 + x_2) + x_2\big) + x_3\big), \tag{6}$$

w której zmienna x_2 termu będzie zmienną związaną, wbrew zastrzeżeniu, że zmienna wolna termu przy podstawieniu nie może przejść na zmienną związaną.

Zauważmy zresztą, że formuła (5) jest mimo to formułą poprawnie zbudowaną. Por. zdanie 4.

Definicja języka elementarnego jest jak widzimy dość skomplikowana. O sensie intuicji jaki leży u podstaw takiego określenia języka można się najlepiej przekonać, zastanawiając się co można za pomocą tego języka opisać. Do tego celu służy następny paragraf.

Streszczenie

Określiliśmy formuły poprawnie zbudowane języka elementarnego. Zmienne predykatywne są formułami poprawnymi. Formuły poprawne możemy łączyć spójnikami zdaniowymi oraz opatrywać kwantyfikatorami. Ponadto można za zmienne wolne podstawiać termy, jeżeli żadna zmienna termu nie stanie się związana. Za zmienne związane nic podstawiać nie wolno. Symbole (alfabet) występujące w formułach zostały starannie wyliczone.

Zadania

1. W poniższych formułach „<" oraz „=" oznaczają predykaty dwuargumentowe, „+" zaś oznacza działanie dwuargumentowe. Zbadać, czy poniższe formuły są poprawnie zbudowane; w formułach poprawnych wyróżnić zmienne wolne i związane:

a) $(x_1 < x_2) \Rightarrow (\mathrm{E}x_1 (x_1 + x_1 = x_2))$,

b) $x_1 < x_2 \Rightarrow \mathrm{E}x_3 (x_1 + x_3 = x_2)$,

c) $x_1 < x_2 \,\&\, x_3 \Rightarrow \mathrm{A}x_1 (x_1 = x_2) \vee x_3$,

d) $\mathrm{A}x_1 \Big(\mathrm{A}x_2 \Big(\mathrm{A}x_3 \Big(\mathrm{A}x_4 \big(((x_1 = x_2) \,\&\, (x_3 < x_4)) \Rightarrow (x_1 + x_3 < x_2 + x_4)\big)\Big)\Big)\Big)$.

2. W formułach a) i d) z poprzedniego zadania usunąć zbędne nawiasy, nie naruszając możliwości jednoznacznego odczytania formuł. Zapisać te spośród nich, które są poprawne, pisząc

zamiast $x_1 < x_2$ wyrażenie $< x_1 x_2$,
zamiast $x_1 = x_2$ wyrażenie $= x_1 x_2$,
zamiast $x_1 + x_2$ wyrażenie $+ x_1 x_2$.

Wypowiedzieć treść tych formuł w języku potocznym.

3. Napisać formuły podane w tekście paragrafu, pisząc wszędzie $Q(x_1, x_2)$ zamiast $x_1 = x_2$ oraz $f(x_1, x_2)$ zamiast x_1+x_2.

4. Wykonując w formule $x_1 = x_2$ dwa podstawienia za zmienne termów oraz dwa razy operację opatrywania kwantyfikatorem, pokazać, że formuła (6) jest poprawnie zbudowana.

5. Uzasadnić, że każdą formułę poprawną można otrzymać, najpierw podstawiając do formuł atomowych termy za zmienne, a potem wykonując operacje typu 2 oraz 3.

§ 24. INTERPRETACJA FORMUŁ POPRAWNYCH

Formuły poprawne zbudowane w języku opisanym w poprzednim paragrafie są pewnymi napisami złożonymi z jakichś symboli. Napisy te nie będą miały żadnej treści, dopóki nie będziemy przypisywali jakiegoś sensu występującym w nich symbolom, tzn. dopóki nie będziemy interpretować formuł. Interpretacja sprowadza się do przypisywania znaczenia zmiennym, zmiennym predykatywnym i symbolom działań występujących w termach. Znaczenie symboli logicznych, spójników zdaniowych i kwantyfikatorów jest ustalone i było wyjaśnione w rozdziałach II i III. Dla pozostałych symboli mamy pewną dowolność interpretowania. Wyjaśnimy to poniżej.

Jeżeli na przykład zmienne interpretujemy jako zmienne przyjmujące wartości będące liczbami naturalnymi, to zmienne predykatywne będziemy uważać za pewne funkcje zdaniowe na zbiorze liczb naturalnych, a termy za pewne działania na liczbach naturalnych. Przy takiej interpretacji, formuły poprawnie zbudowane przedstawiać będą pewne sensowne funkcje zdaniowe lub zdania dotyczące liczb naturalnych. Ciągi symboli nie będące formułami poprawnie zbudowanymi, nie będą przedstawiały żadnych sensownych wypowiedzi.

Na przykład formuła

$$(1) \qquad Q\big(x_1, f(x_2, x_3)\big) \Rightarrow \big(Q(x_1, x_2) \vee Q(x_1, x_3)\big),$$

gdzie f jest symbolem działania dwuargumentowego, Q zaś symbolem dwuargumentowego predykatu jest poprawnie zbudowana. Jeżeli zmienne x_1, x_2, x_3 będziemy interpretowali jako liczby naturalne, f jako działanie mnożenia, predykat zaś $Q(x_1, x_2)$ jako własność mówiącą o podzielności

liczby x_2 przez x_1, to dla liczb naturalnych formułę tę zinterpretujemy następująco,

$$\big(X_1 | (X_2 \cdot X_3)\big) \Rightarrow \big((X_1 | X_2) \vee (X_1 | X_3)\big)$$

(symbol $X_1 | X_2$ czytamy następująco: liczba X_1 dzieli X_2).

Formułę interpretujemy więc jako funkcję zdaniową określaną na liczbach naturalnych mówiącą, że jeżeli liczba dzieli iloczyn, to dzieli jeden z czynników.

Symbole zmiennych, działań i predykatów możemy w tej formule interpretować inaczej (por. zad. 1 tego paragrafu). Zawsze jednak formuła będzie miała jakiś sens.

Natomiast ciąg symboli, który nie jest formułą poprawnie zbudowaną, np.

$$(2) \qquad Q(x_1, x_2) \Rightarrow Q\big(f(x_1, x_3), x_2\big)$$

przy żadnej interpretacji symboli działania f jako działania na liczbach naturalnych i predykatu Q jako związku między liczbami naturalnymi nie będzie miał żadnego sensu.

Omawiając sprawę interpretacji formuł trzeba zauważyć, że w wyniku nie otrzymujemy na ogół wypowiedzi prawdziwych czy fałszywych. Na przykład formuła (2) jest dla jednych trójek liczb naturalnych X_1, X_2, X_3 prawdziwa, dla innych fałszywa. Na przykład dla $X_1 = 2, X_2 = 3, X_3 = 5$ wypowiedź (2) jest prawdziwa, podobnie dla $X_1 = 2, X_2 = 2, X_3 = 1$. Mówimy, że formuła jest spełniona przy takiej interpretacji jak powyżej. Ale na przykład dla $X_1 = 2, X_2 = 4, X_3 = 3$ wypowiedź (2) jest fałszywa. Mówimy, że formuła ta nie jest spełniona przy takiej interpretacji.

Interpretując formułę otrzymamy na ogół w wyniku nie zdanie, ale funkcję zdaniową. Zdanie otrzymamy wtedy, gdy formuła nie zawiera zmiennych wolnych.

Na przykład interpretując formułę

$$\mathrm{A}X_1\Big(\mathrm{A}X_2\Big(\mathrm{A}X_3\big(Q\big(X_1, f(X_2, X_3)\big) \Rightarrow \big(Q(X_1, X_2) \vee Q(X_1, X_3)\big)\big)\Big)\Big)$$

tak jak formułę (1) otrzymamy zdanie fałszywe:

dla każdych liczb naturalnych X_1, X_2, X_3:

$$(3) \qquad (X_1 | X_2 \cdot X_3) \Rightarrow \big((X_1 | X_2) \vee (X_1 | X_3)\big).$$

W naszym przypadku formuły (1) zbiór N liczb naturalnych, z określoną funkcją zdaniową $X|Y$ (liczba naturalna X dzieli liczbę naturalną Y oraz działaniem „·" mnożenia liczb naturalnych stanowi model formuły, to znaczy, że formuła jest prawdziwa przy podanej interpretacji Q oraz f.

Podobnie określamy pojęcie *modelu zbioru formuł*. Modelem zbioru formuł zawierających zmienne predykatywne $P, Q, R, \ldots$ oraz działania $f, g, k, \ldots$ jest zbiór N jakichś przedmiotów, w którym określane są konkretne funkcje zdaniowe $P(X), Q(X, Y), R(X, Y, Z), \ldots$, dla każdych X, Y, Z ze zbioru N oraz działanie $F(X), G(X, Y), H(X, Y, Z)$, tak że przy podanej interpretacji prawdziwe są wszystkie formuły.

Formuły, będące niczym innym jak ciągami napisów, nabierają w modelu określonego sensu, opisują pewne fakty matematyczne odnoszące się do funkcji zdaniowych i działań.

Zauważmy jeszcze, że w podanym przez nas języku teorii nie możemy opisać wszystkich faktów matematycznych, nie możemy wypowiedzieć np. zdań, że dla każdego zbioru elementów coś zachodzi, dla każdego termu coś zachodzi, dla każdego predykatu coś zachodzi itd.

Związane to jest z tym, że nie możemy używać kwantyfikatorów wiążących zmienne przebiegające np. zbiór wszystkich predykatów, czy też termów.

Teorie o takim języku, jak podaliśmy nazywają się teoriami elementarnymi lub pierwszego rzędu, formuły zaś formułami elementarnymi. Teorie takie nie nadają się wprawdzie do opisu całej matematyki, lecz stanowią dostatecznie dobry i prosty przykład sformalizowanych teorii matematycznych służący do ilustracji zagadnień związanych z formalizacją matematyki.

Streszczenie

Wyjaśniliśmy jak możemy interpretować formuły poprawne jako pewne sensowne wypowiedzi matematyczne. Teorie w podanym przez nas języku w którym kwantyfikatory stosuje się tylko do zmiennych $x_1, x_2, \ldots$ przedstawiających elementy, nazywają się teoriami elementarnymi.

Zadania

1. Podać dla formuły (1) funkcję zdaniową otrzymaną przez interpretację f jako działania „+" dodawania liczb naturalnych, predykat zaś Q jako „=" równości liczb naturalnych.

2. Uzasadnić, że zbiór N_1 liczb parzystych większych od zera z funkcją zdaniową $X = Y$ (liczba X równa się liczbie Y) oraz działaniem „·" mnożenia nie jest modelm formuły (1), gdyż formuła nie jest prawdziwa przy takiej interpretacji.

§ 25. TWIERDZENIA I DOWODY W TEORIACH SFORMALIZOWANYCH

W dalszym ciągu zajmować się będziemy tylko teoriami elementarnymi.

W poprzednich paragrafach omówiliśmy język matematycznych teorii sformalizowanych i powiedzieliśmy co to są formuły poprawnie zbudowane.

Mając dany język, mówimy o teorii wtedy, gdy spośród wszystkich formuł wyróżniona jest pewna klasa formuł zwanych twierdzeniami teorii.

W teoriach dedukcyjnych (a takimi są matematyczne teorie sformalizowane) klasę twierdzeń określamy przez podanie aksjomatów teorii, aksjomatów logicznych oraz reguł dedukcji.

Aksjomaty teorii są to wyjściowe twierdzenia teorii podane w postaci formuł poprawnie zbudowanych. *Reguły dedukcji* są to reguły prowadzące od formuł do formuł. Pozwalają one z jednych twierdzeń teorii otrzymywać inne twierdzenia teorii. W szczególności przy zadanych aksjomatach teorii i regułach dedukcji, zbiór twierdzeń teorii jest to zbiór tych formuł, które można otrzymać z aksjomatów teorii przez stosowanie do nich skończonej ilości razy reguł dedukcji. Mówiąc prościej twierdzenia teorii są to te formuły, które można dowieść z aksjomatów.

Na ogół przyjmuje się w teoriach matematycznych cztery następujące reguły dedukcji:

1. Regułę pozwalającą uznać za twierdzenie teorii formułę, która jest szczególnym przypadkiem aksjomatu logicznego.

2. Regułę odrywania pozwalającą z dwóch formuł poprawnych Φ oraz $\Phi \Rightarrow \Psi$ otrzymać formułę Ψ.

3. Regułę podstawiania pozwalającą w formule

(1) $$\Phi(x_1, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_s)$$

o zmiennych wolnych $x_1, \ldots, x_s$ podstawić za zmienną x_j term $f(x_{a_1}, \ldots, x_{a_s})$ i otrzymać formułę

$$\Phi(x_1, \ldots, x_{j-1}, f(x_{a_1}, \ldots, x_{a_s}), x_{j+1}, \ldots, x_s) \tag{2}$$

pod warunkiem, że po wykonaniu podstawienia żadna zmienna nie przejdzie na zmienną związaną. Warto zauważyć, że reguła podstawienia nie jest jedną regułą, lecz pewnym schematem. Mianowicie dla ustalonej zmiennej x_j i ustalonego termu

$$f(x_{a_1}, \ldots, x_{a_s})$$

otrzymujemy jedną regułę prowadzącą od dowolnej formuły (1) do formuły postaci (2).

4. Regułę uogólniania, prowadzącą od formuły $\Phi(x)$ do formuły $\mathrm{A}x\,\Phi(x)$.

Aksjomaty dzielimy na aksjomaty logiczne i aksjomaty teorii.

Aksjomaty logiczne są to aksjomaty F_1-F_4 rachunku kwantyfikatorów, wymienione w rozdziale III § 21 str. 80 Są to:

F_1. Wszystkie tautologie rachunku zdań.

F_2. Tautologie postaci

$$\vdash \mathrm{A}x\big(A \Rightarrow B(x)\big) \Rightarrow \big(A \Rightarrow \mathrm{A}x\,B(x)\big),$$

$$\vdash \mathrm{A}x\big(B(x) \Rightarrow A\big) \Rightarrow \big(\mathrm{E}x\,B(x) \Rightarrow A\big).$$

F_3. $\mathrm{A}x\,B(x) \Rightarrow B(x)$.

F_4. $\mathrm{E}x\big(B(x) \vee \sim B(x)\big)$.

Aksjomatami teorii może być jakikolwiek zbiór formuł poprawnych teorii. Aksjomaty te zwane też często *aksjomatami specyficznymi* określają własności pojęć pierwotnych występujących w tej teorii, tzn. własności predykatów i funktorów (działań) występujących w teorii. W szczególności gdy brak aksjomatów specyficznych (a w języku teorii — symboli działań, otrzymujemy rachunek kwantyfikatorów opisany w rozdziale III).

Podamy teraz pojęcie dowodu w teorii sformalizowanej. Ideą dowodzenia jest wyjście od jakiegoś zbioru formuł A, zwanego *zbiorem aksjomatów teorii* oraz od aksjomatów logicznych i otrzymywanie twierdzeń teorii przez stosowanie reguł dedukcji. Formułę Φ uznamy za twierdzenie, jeżeli da się ją otrzymać z aksjomatów teorii oraz z aksjomatów logicznych przez zastosowanie skończoną ilość razy reguł dedukcji.

Opisaną ideę otrzymywania twierdzeń teorii najłatwiej wypowiedzieć w sposób bardzo ścisły, wprowadzając pojęcie dowodu. Ciąg formuł

$$\Phi_1, \ldots, \Phi_n \tag{3}$$

nazywamy *dowodem* formuły Φ, jeśli $\Phi_n = \Phi$, gdzie liczbę n nazywamy *długością dowodu*, oraz jeśli dla każdej formuły Φ_i $(1 \leqslant i \leqslant n)$ zachodzi jeden z poniższych warunków:

0′ Φ_i jest aksjomatem teorii;

1′ Φ_i jest szczególnym przypadkiem aksjomatu logicznego;

2′ istnieją dwie formuły Φ_j i Φ_k, gdzie $1 \leqslant j < i$ oraz $1 \leqslant k < i$ takie, że Φ_k ma następującą postać:

$$\Phi_j \Rightarrow \Phi_i;$$

3′ istnieje formuła Φ_l, $l < i$, stojąca w ciągu (3) przed formułą Φ_i, taka że

$$\Phi_l = \Phi_l(x_1, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots x_s)$$

zaś

$$\Phi_i = \Phi_l(x_1, \ldots, x_{j-1}, f(x_{a_1}, \ldots, x_{a_s}), x_{j+1}, \ldots, x_s),$$

gdzie $f(x_{a_1}, \ldots, x_{a_s})$ jest termem;

4′ istnieje formuła Φ_l, $l < i$, taka że Φ_i jest postaci $\mathrm{A}x\,\Phi_l(x)$, gdzie x jest jakąś zmienną indywidualną.

Oczywiście z określenia dowodu widać natychmiast, że punktom 1′, 2′, 3′, 4′ odpowiada stosowanie reguł 1, 2, 3, 4.

Zbiór twierdzeń jakiejś teorii o zbiorze aksjomatów A, to zbiór tych formuł poprawnie zbudowanych, dla których istnieje dowód. Zauważmy, że w myśl tego określenia każdy aksjomat jest twierdzeniem teorii. Istotnie dowód aksjomatu Φ jest to, jak łatwo sprawdzić, jednowyrazowy ciąg złożony z tej formuły Φ.

Jeśli jako zbiór aksjomatów A przyjmiemy zbiór pusty (nie zawierający żadnego elementu), wówczas zbiorem twierdzeń takiej teorii będzie zbiór tautologii rachunku kwantyfikatorów.

Zbiór twierdzeń sformalizowanej teorii matematycznej jest w ten sposób poprawnie określony. Język teorii i reguły pozwalające otrzymywać twierdzenia teorii zostały precyzyjnie określone. Wiadomo co to są formuły poprawnie zbudowane i wiadomo co to jest dowód twierdzenia.

Powstaje pytanie czy z podanych określeń potrafimy wywnioskować, które formuły teorii są jej twierdzeniami. Podana została definicja twierdzenia teorii. Jeżeli dla formuły Φ istnieje jej dowód z aksjomatów A, to formuła jest twierdzeniem teorii. Jeżeli takiego dowodu nie ma, to formuła nie jest twierdzeniem teorii. Nie wynika stąd jednak, że potrafimy dla każdej formuły stwierdzić efektywnie czy jest ona twierdzeniem teorii, czy nie. Przeprowadzenie dowodów jest czynnością trudną. Wprawdzie kroki postępowania dowodowego, tzn. reguły dedukcji zostały precyzyjnie opisane, tak że można je stosować w sposób mechaniczny, jednak na ogół nie widać żadnej ogólnej metody pozwalającej znajdować dowody danej formuły. Fakt, że po pewnych próbach dowodu formuły nie uzyskaliśmy, nie pozwala wnioskować, że jej dowodu nie ma. Być może korzystając z reguł dedukcji w innej kolejności dowód znajdziemy, a być może formuła po prostu nie jest twierdzeniem teorii.

Dla większości teorii sformalizowanych nie ma żadnej ogólnej metody stwierdzenia czy dowolna formuła jest twierdzeniem teorii, czy też nie. Sytuacja jest podobna jak w rachunku kwantyfikatorów, gdzie nie było metody rozstrzygania czy formuła jest tautologią, czy też nie. Teorie matematyczne są na ogół nierozstrzygalne, tak jak rachunek kwantyfikatorów.

Streszczenie

Podaliśmy reguły dedukcji matematycznych teorii sformalizowanych: 1. regułę uznania za twierdzenie każdej formuły, która jest szczególnym przypadkiem tautologii logicznej. 2. regułę odrywania, 3. reguły podstawiania. 4. regułę uogólniania. Podaliśmy pojęcie dowodu teorii i twierdzenia teorii. Zwróciliśmy uwagę na fakt, że mimo precyzyjnego określenia kroków dowodowych na ogół nie ma metody rozstrzygania czy formuła jest twierdzeniem teorii, czy też nie.

Zadania

1. Uzasadnić, że jeżeli ciąg $\Phi_1, \ldots, \Phi_n$ jest dowodem formuły Φ, to formułę da się uzyskać za pomocą n-krotnego stosowania reguł 1, 2, 3, 4.

2. a) Z jakich tautologii należy korzystać, żeby udowodnić, że jeżeli formuły Φ oraz Ψ są twierdzeniami teorii, to twierdzeniami teorii są formuły: $\Phi \vee \Psi$, $\Phi \,\&\, \Psi$, $\Phi \Rightarrow \Psi$ oraz $\Phi \equiv \Psi$?

b) Z jakiej tautologii rachunku kwantyfikatorów i jakich reguł należy skorzystać, aby udowodnić, że jeżeli formuła $Ax_1\Phi(x_1)$ jest twierdzeniem teorii, to formuła $\Phi(x_1)$ jest twierdzeniem teorii?

c) Udowodnić, że jeżeli formuła $Ax_1\Phi(x_1)$ jest twierdzeniem teorii i zmienna x_2 nie występuje w formule, to formuła $Ax_2\Phi(x_2)$ jest twierdzeniem teorii.

d) Niech $A_1, A_2, \ldots, A_k$ będą aksjomatami teorii. Podać pełny dowód formuły $A_1 \,\&\, A_2 \,\&\, \ldots \,\&\, A_n$, jaka będzie jego długość? Przypominamy, że długością dowodu $\Phi_1, \Phi_2, \ldots \Phi_n$ nazywamy liczbę występujących w nim formuł, tj. n.

Wskazówka: Skorzystać z tautologii:

$$p_1 \Rightarrow \Big(p_2 \Rightarrow \Big(\ldots \big(p_n \Rightarrow p_1 \,\&\, (p_2 \,\&\, (\ldots \,\&\, p_n) \ldots)\big)\Big) \ldots\Big).$$

3. a) Uzasadnić, że jeżeli liczba zmiennych predykatywnych i liczba działań jest skończona, to wyrażenia poprawne teorii można ustawić w ciąg.

b) Uzasadnić, że w tym wypadku można ustawić w ciąg wszystkie dowody.

c) Uzasadnić, że można w ten sposób ustawić w ciąg wszystkie twierdzenia teorii.

4. Teoria, w której z każdej pary zdań Φ oraz $\sim \Phi$ dokładnie jedno jest twierdzeniem, nazywa się *teorią zupełną*. Opierając się na zadaniu 3, podać metodę rozstrzygania czy formuła jest twierdzeniem teorii, czy też nie.

Wskazówka: W ciągu dowodów znajdziemy albo dowód formuły Φ, albo dowód formuły $\sim \Phi$.

5. *Teoria* nazywa się *sprzeczna*, jeżeli istnieje taka formuła Φ, że Φ oraz $\sim \Phi$ są twierdzeniami teorii.

Korzystając z tautologii rachunku zdań $(p \,\&\, \sim p) \Rightarrow q$, udowodnić, że w teorii sprzecznej każda formuła poprawna jest twierdzeniem teorii.

§ 26. MODELE TEORII ELEMENTARNYCH

Teorie w języku opisanym w paragrafie 23 tego rozdziału i o regułach dedukcji opisanych w poprzednim paragrafie nazywamy *teoriami elementarnymi* (lub *teoriami pierwszego rzędu*).

Nazwa ta pochodzi stąd, że operacje kwantyfikatora możemy stosować tylko do zmiennych elementarnych, a nie możemy ich stosować do predykatów czy też symboli działań. Predykaty i działania są w teorii ustalone w tym sensie, że ich własności (czy lepiej powiedzieć możliwości interpretacji) wyrażane są przez aksjomaty specyficzne. Stwierdzenie to wymaga

bliższego wyjaśnienia. Predykaty jak i funktory (ogólniej mówiąc termy) w teorii sformalizowanej są pewnymi symbolami. Aksjomaty mówią jak możemy interpretować predykaty i termy (§ 24 tego rozdziału). Możemy korzystać tylko z takich interpretacji, przy których aksjomaty specyficzne są prawdziwe.

Modelem teorii o zbiorze A aksjomatów specyficznych nazywamy model zbioru formuł należących do A (zob. str. 94), a więc modelem teorii będzie model zbioru aksjomatów specyficznych teorii.

Model taki ustala taką interpretację zmiennych predykatywnych i funktorów (symboli działań) występujących w aksjomatach specyficznych, przy której aksjomaty będą funkcjami zdaniowymi prawdziwymi.

Aksjomaty logiczne są tautologiami rachunku kwantyfikatorów, a więc będą prawdziwe przy każdej interpretacji, nie tylko przy podanej w modelu. Reguły dedukcji 1-4 teorii prowadzą od funkcji zdaniowych prawdziwych przy jakiejś interpretacji do funkcji zdaniowych prawdziwych. Ponieważ każde twierdzenie teorii uzyskuje się z aksjomatów logicznych i aksjomatów dedukcyjnych, a więc przy żądanej interpretacji w modelu wszystkie twierdzenia teorii będą funkcjami zdaniowymi prawdziwymi.

Teoria nazywa się *niesprzeczna*, jeżeli wśród jej twierdzeń nie występuje para formuł Φ oraz $\sim\Phi$. W przeciwnym przypadku *teoria* nazywa się *sprzeczna*. Teorie sprzeczne na pewno modeli nie mają. Teorie niesprzeczne mają modele.

Reguły 1-4 dedukcji prowadziły od formuł prawdziwych do formuł prawdziwych. Aksjomaty logiczne były formułami prawdziwymi. Inaczej nieco ma się sprawa z aksjomatami specyficznymi teorii. Każda teoria elementarna ma wiele modeli. Na przykład w teorii o jednym predykacie Q dwuargumentowym i jednym aksjomacie (1) (podanym w paragrafie 24), możemy interpretować zmienne $x_1, x_2, \ldots$ jako np. liczby naturalne należące do jakiegoś zbioru N liczb naturalnych. Predykat Q możemy interpretować jako np. równość (tzn. $Q(a_1, a_2)$ zachodzi wtedy i tylko wtedy, gdy $a_1 = a_2$). Ale równie dobrze aksjomat (1) będzie spełniony, jeżeli zinterpretujemy predykat Q, jako różność ($Q(a_1, a_2)$ zachodzi wtedy i tylko wtedy, gdy $a_1 \neq a_2$).

Przykład ten nieco drastyczny pokazuje, że teoria podana w tym przykładzie jest bardzo uboga. Nie chodzi tylko o to, że ma mało pojęć pierwotnych i jedno działanie. Ma ona mało aksjomatów, co dopuszcza wiele interpretacji pojęć (jak widzimy w omówionym przykładzie nawet w sposób diametralnie przeciwny).

Wszystkie teorie elementarne mają tę cechę. Zawsze możemy pojęcia występujące w teorii interpretować na wiele różnych sposobów — matematycy mówią, że teorie elementarne mają wiele modeli. Cecha ta wcale nie jest wadą teorii elementarnych. Wiele interpretacji dopuszcza wiele zastosowań. Każde twierdzenie teorii będzie spełnione przy każdej interpretacji, przy której spełnione są aksjomaty.

Pozostaje jeszcze kwestia prawdziwości twierdzeń teorii elementarnych.

Należy tu zwrócić uwagę, że aksjomaty specyficzne teorii mogą być dowolnymi formułami poprawnymi. Przez ich wybór wybieramy sens jaki chcemy przypisywać pojęciom pierwotnym — predykatom i działaniom teorii występujących w aksjomatach specyficznych. Stwierdzenie, że aksjomaty specyficzne teorii mają sens prawdziwy, znaczy tyle, że predykatom i działaniom w nim występującym możemy przypisywać tylko taki sens, przy którym aksjomaty będą„ prawdziwe" w intuicyjnym rozumieniu tego słowa.

Twierdzenia teorii są dzięki definicji modelu zdaniami prawdziwymi, tzn. są prawdziwe w każdym modelu.

W przypadku teorii zachodzi również twierdzenie odwrotne. Każda formuła prawdziwa, tzn. prawdziwa w każdym modelu teorii jest twierdzeniem teorii. Matematycy mówią, że *teorie elementarne są pełne*[1]. Fakt ten nie jest tak oczywisty jak by się mogło na pierwszy rzut oka zdawać i dla innych teorii nie jest prawdziwy (por. § 27). Twierdzenie o pełności teorii elementarnych mówi, że w przypadku teorii elementarnych reguły dedukcji są dostatecznie potężnym narzędziem by za ich pomocą można było udowodnić (wywnioskować z aksjomatów) każdy prawdziwy fakt dający się wyrazić w teorii. Wynik ten świadczy jednak o ubóstwie języka i pojęć teorii elementarnych. Kwestie te są omówione w następnym paragrafie.

Streszczenie

Zdefiniowaliśmy pojęcie teorii elementarnych. Omówiliśmy rolę aksjomatów specyficznych i związaną z nimi sprawę interpretacji pojęć teorii —

[1] Należy odróżniać to pojęcie od pojęcia zupełności podanego na str. 99.

modelu teorii. Teorie elementarne dopuszczają wiele interpretacji. Podaliśmy pojęcie formuły prawdziwej, tzn. formuły prawdziwej w każdym modelu teorii. Twierdzenia są formułami prawdziwymi teorii. W przypadku teorii elementarnych każda formuła prawdziwa jest twierdzeniem teorii.

Zadania

1. Dla teorii o jednym predykacie $Q(x_1, x_2)$ i aksjomacie

$$(*) \qquad Q(x_1, x_2) \Rightarrow Q(x_2, x_1)$$

interpretujemy zmienne jako liczby naturalne, predykat Q zaś jako związek $Q(a_1, a_2)$, który zachodzi wtedy i tylko wtedy, gdy $a_1 = a_2+1$ lub $a_2 = a_1+1$.

a) Czy prawdziwy jest w tym modelu aksjomat (*)?

b) Udowodnić, że formuła $Ax_1(Q(x_1, x_1))$ nie jest twierdzeniem teorii.

2. Uzasadnić, że teoria z zadania 1 jest niezupełna. Rozpatrzyć w tym celu formułę

$$(**) \qquad Ex_1(Q(x_1, x_1)).$$

i interpretować Q raz jako relację równości, raz jako relację różności. Przy pierwszej interpretacji formuła będzie prawdziwa, przy drugiej zaś nie. Pozwala to wywnioskować (dlaczego?), że ani formuła (**) ani jej negacja nie są twierdzeniami teorii.

3. Jeżeli formuła Φ jest twierdzeniem teorii o aksjomatach $A_1, \ldots, A_n$, to formuła

$$(A_1 \,\&\, (A_2 \,\&\, (\ldots \,\&\, A_n) \ldots)) \Rightarrow \Phi$$

jest tautologią logiczną.

Twierdzenie to nazywa się *twierdzeniem o dedukcji*. Udowodnić twierdzenie o dedukcji w przypadku, gdy w dowodzie korzystamy z reguł 1′ i 2′ ze str. 97.

Wskazówka: Dowód przeprowadzić indukcyjnie względem długości dowodu n.

4. Udowodnić, że teoria sprzeczna nie może mieć żadnych modeli.

Wskazówka. Jeżeli Φ oraz $\sim \Phi$ są twierdzeniami teorii o aksjomatach $A_1, \ldots A_n$, to implikacje

$$(A_1 \,\&\, (\ldots \,\&\, A_n)) \Rightarrow \Phi$$

oraz

$$(A_1 \,\&\, (\ldots \,\&\, A_n)) \Rightarrow \sim \Phi$$

są tautologiami. Implikacja

$$(A_1 \,\&\, (\ldots \,\&\, A_n)) \Rightarrow (\Phi \,\&\, \sim \Phi)$$

będzie również tautologią. A więc tautologią będzie formuła

$$\sim (A_1 \,\&\, (\ldots \,\&\, A_n)).$$

Wywnioskować stąd, że aksjomaty $A_1, \ldots, A_n$ nie mogą być jednocześnie spełnione w modelu.

§ 27. TEORIE MATEMATYCZNE

Ten końcowy fragment poświęcony jest omówieniu wiadomości z poprzednich paragrafów i ich powiązaniu z dalszymi fragmentami książki. Ponadto omówione są zagadnienia związane z teoriami nieelementarnymi i innymi teoriami sformalizowanymi.

Zajmując się matematyką, spotykamy się z dwoma rodzajami pojęć. Pierwsze to pojęcia czysto matematyczne, drugie to pojęcia logiczne.

Pojęciem matematycznym jest np. pojęcie liczby naturalnej. Bada się zbiory liczb naturalnych, własności działań takich jak dodawanie czy mnożenie, własności stosunków między liczbami naturalnymi, takich jak stosunek równości, mniejszości, podzielności i inne.

Drugi rodzaj pojęć, które już omówiliśmy, to pojęcia pozamatematyczne — pojęcia logiczne. W wypowiedziach matematycznych spotykamy się ze zwrotami „jeżeli ..., to", „nieprawda, że...", „dla każdej liczby naturalnej ...", „istnieje element taki, że ...". Zwroty te, to jak wiemy spójniki zdaniowe i kwantyfikatory. Bez ich pomocy wysłowienie twierdzeń matematycznych byłoby niemożliwe.

Badaniem własności spójników zdaniowych i kwantyfikatorów oraz praw posługiwania się nimi zajmują się teorie logiczne zwane rachunkiem zdań i rachunkiem kwantyfikatorów, omówione w rozdziale II i III. Twierdzenia tych teorii noszą nazwę tautologii logicznych. Tautologie logiczne są to po prostu prawa logicznego rozumowania podające jakie wypowiedzi są prawdziwe nie z powodu takiej czy innej treści zdań w nich występujących, lecz z powodu samej struktury wypowiedzi.

Własności pojęć matematycznych opisane są przez aksjomaty specyficzne. Z aksjomatów drogą stosowania znanych nam już reguł dedukcji otrzymujemy twierdzenia teorii matematycznych.

Na ogół symbolom matematycznej teorii sformalizowanej możemy przypisywać sens na wiele różnych sposobów — matematycy mówią, że możemy budować wiele modeli jakiejś teorii sformalizowanej. Przykłady modeli takich teorii podane są w rozdziałach VII i VIII. Pewne formuły teorii, a wśród nich z konieczności aksjomaty (a więc i twierdzenia) będą prawdziwe we wszystkich modelach. Formuły takie zgodnie z naszą intuicją nazywamy zdaniami prawdziwymi teorii.

Mamy więc dwa pojęcia: pojęcie twierdzenia teorii i pojęcie zdania prawdziwego teorii. Jaki zachodzi między nimi związek?.

Twierdzenia teorii są jak wiemy zdaniami prawdziwymi teorii (por. par. 26 tego rozdziału). Zachodzi pytanie czy są one wszystkimi zdaniami prawdziwymi teorii. Jak wiemy z par. 26 tego rozdziału dla teorii elementarnych tak jest. Twierdzenia pokrywają się ze zdaniami prawdziwymi.

Na ogół jednak, jeżeli teoria jest dość bogata — jeżeli można w niej zainterpretować dostatecznie dużo pojęć matematycznych, to reguły dedukcji nie są (i nie mogą być) dość mocne by można było podać taką aksjomatykę, której każde zdanie prawdziwe teorii dałoby się uzyskać jako twierdzenie teorii(1).

W tym ostatnim zdaniu winny ulec wyjaśnieniu słowa „dość bogata". Bogactwo teorii polega na jej języku. Teorie elementarne są ubogie, gdyż ich język jest ubogi. Nie pozwala na przykład opisać zdań: „dla każdego predykatu $P(x)$" czy też „dla każdego działania $f(x_1, \ldots, x_n)$", gdyż kwantyfikatorami można działać tylko na zmienne elementarne przebiegające przedmioty, a nie predykaty czy też działania.

Sformalizowane teorie matematyki, w których języku występują formuły typu

$$\begin{aligned} &\mathrm{A}P_i\Phi(x_1, \ldots, x_n, P_1, \ldots, P_s), \\ &\mathrm{E}P_i\Phi(x_1, \ldots, x_n, P_1 \ldots, P_s), \end{aligned} \tag{1}$$

gdzie Φ jest pewną formułą zawierającą zmienne $x_1, \ldots, x_n$ przebiegające przedmioty, i zmienne $P_1, \ldots, P_s$ przebiegające predykaty nazywają się *teoriami nieelementarnymi*. Formuły postaci (1) nazywają się *formułami nieelementarnymi* (ściślej mówiąc *formułami nieelementarnymi drugiego rzędu*). Odnośnie szczegółów odsyłamy czytelnika do literatury specjalnej: Mostowskiego, Grzegorczyka [1961], Rasiowej i Sikorskiego.

(1) Wynik ten stanowi treść słynnego twierdzenia Gödla. Omówienie go choćby pobieżnie wymagałoby oddzielnego rozdziału. Nie uczynimy tego, gdyż te kwestie wypłynęły na marginesie naszych rozważań. Odsyłamy czytelnika do bardziej specjalnych opracowań np. E. Nagel i J. R. Newman, względnie do podręczników logiki, Mostowskiego, Grzegorczyka [1961], Kleene'a [1952]. Twierdzenie Gödla jest omówione również w wielu książkach związanych z teorią maszyn matematycznych. W cytowanych książkach, jak również Arbiba, Głuszkowa, jak też Rasiowej i Sikorskiego znajdzie czytelnik również obszerne wiadomości dotyczące innych zagadnień omawianych w tym rozdziale.

Teorie nieelementarne mają język na tyle bogaty, że można nim opisać całą matematykę, arytmetykę liczb naturalnych, teorię liczb rzeczywistych, analizę matematyczną, geometrię i inne działy matematyki.

Oprócz teorii sformalizowanych matematyki bada się również zarówno w całej matematyce, jak i w innych gałęziach nauk jej pokrewnych np. lingwistyce matematycznej, inne teorie sformalizowane oparte na innych regułach dedukcji nie związanych z prawami logiki.

Opisane w rozdziale IX systemy Posta i Thuego podpadają również pod podane przez nas określenie teorii sformalizowanej. Wyrażenia poprawne w tych systemach (teoriach) można interpretować jako elementy pewnych półgrup (por. rozdział VIII).

System Chomskiego opisany w paragrafie 52 rozdziału IX również podpada pod określenie teorii sformalizowanej. Twierdzenia tej teorii można interpretować jako zdania poprawne gramatycznie w jakimś uproszczonym języku naturalnym.

Wymienione powyżej teorie są teoriami dedukcyjnymi i to nawet sformalizowanymi, ale nie są teoriami stricto sensu matematycznymi, choć są zmatematyzowane i posługują się językiem właściwym matematyce. Systemy Posta i Thuego choć stworzone przez matematyków nie są matematycznymi teoriami dedukcyjnymi, gdyż reguły dedukcji używane w tych systemach nie są oparte o reguły wnioskowania logicznego, a tylko o pewne reguły przekształcenia wyrażeń teorii. Teoria Chomskiego jest teorią lingwistyczną (lingwistyki matematycznej).

W teoriach tych nie ma potrzeby mówić o prawdzie czy fałszu twierdzeń, gdyż nie interpretujemy jej formuł jako wypowiedzi. W związku z tym ważną sprawą jest problem rozstrzygalności teorii sformalizowanych.

Streszczenie

W paragrafie tym streściliśmy pokrótce problemy formalizacji teorii, rozpatrywane w poprzednich paragrafach. Wspomnieliśmy również o teoriach nieelementarnych oraz o wyniku Gödla. Ponadto wspomnieliśmy o przykładach teorii sformalizowanych, o regułach dedukcji nie związanych z prawami logiki.

Rozdział 5

RACHUNEK ZBIORÓW

W matematyce badamy przeróżne zbiory: zbiory punktów płaszczyzny, zbiory jakichś funkcji, figur geometrycznych lub innych abstrakcyjnych elementów. Z tego względu pojęcie zbioru jest bardzo ważnym pojęciem matematyki.

Odróżnia się przy tym dwa pojęcia: pojęcie *elementu* i pojęcie *zbioru*. Zbiór składa się z elementów, co matematycznie wyrażamy mówiąc, że między elementami a zbiorem zachodzić może stosunek należenia, co piszemy $x \in X$, a czytamy „element x należy do zbioru X". Wypowiedź ta znaczy, że element x jest jednym z elementów, z których składa się zbiór X.

Od elementów, z których składa się zbiór, nie żądamy by były jednorodne lub jednakowego rodzaju. Jednakowo jest do pomyślenia zbiór kamyków, jak i zbiór złożony z kota, psa, trójkąta, boków trójkąta i pomarańczy, a więc przedmiotów najzupełniej różnej natury.

Żądamy tylko, by dla elementów każdego rozpatrywanego zbioru było dobrze określone czy element należy do tego zbioru, czy nie. Ponadto zbiory rozpatruje się niezależnie od porządku elementów w nich występujących: dwa zbiory są identyczne wtedy i tylko wtedy, gdy mają te same elementy.

Omówione tutaj pokrótce pojęcie zbioru jest pojęciem intuicyjnym jednak nieoględne posługiwanie się nim prowadzi do wielu niebezpieczeństw (zob. rozdział I § 3). Dlatego pojęcie zbioru zostało zaksjomatyzowane. W wieku XX matematyk włoski E. Zermelo stworzył aksjomatyczną teorię zbiorów — zaksjomatyzował pojęcie zbioru. Badania nad tą teorią kontynuowane są do dziś. Z uwagi na specjalny charakter przedmiotu, aksjomatyki pojęcia zbioru nie będziemy rozpatrywać w naszej książce, odsyłając czytelnika, którego ten przedmiot mógłby zainteresować do książek specjalnych: Kuratowskiego i Mostowskiego, Fraenkel'a lub do podręcznika Kuratowskiego względnie Słupeckiego i Borkowskiego lub książeczki Sierpińskiego.

W rozdziale tym zajmiemy się natomiast małym fragmentem teorii mnogości, dającym się przedstawić w oparciu o opisane intuicyjnie pojęcie zbioru, tak zwanym *rachunkiem zbiorów* badającym różne operacje jakie można wykonywać na zbiorach i związki między tymi operacjami a rachunkiem zdań czy rachunkiem kwantyfikatorów. Rozumienie rachunku zbiorów jest niesłychanie ważne dla rozumienia nie tylko różnych dalszych fragmentów tej książki, lecz również do rozumienia matematyki.

§ 28. ZBIORY JAKO WŁASNOŚCI ELEMENTÓW

Weźmy jakiś zbiór elementów X, i rozpatrzmy wszystkie zbiory złożone z elementów tego zbioru. Zbiory te nazywać będziemy *podzbiorami* zbioru X. Dla ustalenia uwagi i uniknięcia pomyłek, zbiór X złożony z wszystkich elementów przez nas rozpatrywanych nazywać będziemy *zbiorem pełnym*, zbiory zaś przez nas rozpatrywane, tzn. podzbiory zbioru pełnego nazywać będziemy krótko *zbiorami*. Zbiór złożony z elementów $a_1, \ldots, a_n$ oznaczać będziemy symbolem $\{a_1, \ldots, a_n\}$.

Jeżeli na przykład rozpatrywać będziemy elementy 1, 2, 3, 4, 5, to zbiorem pełnym X będzie $X = \{1, 2, 3, 4, 5\}$.

Zbiorami przez nas rozpatrywanymi będą np. następujące podzbiory: $\{3\}$ (podzbiór złożony z jednego elementu), $\{1, 2, 5\}$, $\{2, 4\}$, $\{2, 3, 5\}$ i inne podzbiory X. Zbiór elementów $\{1, 5, 7, 9\}$ nie będzie w naszym rozumieniu zbiorem, który możemy rozpatrywać, gdyż zawiera elementy nie należące do zbioru pełnego, mianowicie 7 i 9.

W takim rozumieniu zbiory są własnościami uprzednio ustalonych przedmiotów zbioru pełnego. Każdemu zbiorowi odpowiada pewna własność tych przedmiotów, każdej zaś własności pewien zbiór.

Na przykład każdemu elementowi zbioru $A = \{2, 3, 5\}$ będzie przysługiwać własność: być liczbą pierwszą; spośród wszystkich elementów zbioru pełnego tylko tym elementom zbioru A ta własność będzie przysługiwać.

Na odwrót, własności: być liczbą podzielną przez dwa, określonej dla zbioru pełnego X będzie odpowiadać zbiór wszystkich tych elementów zbioru pełnego, które tę własność mają, czyli zbiór $B = \{2, 4\}$.

Niech A będzie podzbiorem zbioru pełnego X, $P(x)$ zaś pewną funkcją zdaniową zmiennej x określoną dla x należących do X. Mówimy, że funkcja zdaniowa $P(x)$ wyznacza zbiór A, jeżeli:

$$(1) \qquad \mathrm{A}x\big(P(x) \equiv x \epsilon A\big).$$

Słowami możemy powiedzieć, że funkcja zdaniowa $P(x)$ określona w zbiorze X wyznacza podzbiór A zbioru pełnego, złożony z tych wszystkich przedmiotów, które spełniają $P(x)$. Równoważność (1) mówi, że dla elementu $a \epsilon X$, jeżeli $P(a)$ jest zdaniem prawdziwym, to $a \epsilon A$, i na odwrót, jeżeli $P(a)$ jest zdaniem fałszywym, to nieprawda że $a \epsilon A$ (to ostatnie zapiszemy w skrócie $a \notin A$ i czytamy, że przedmiot a nie należy do zbioru A).

Oczywiście wiele funkcji zdaniowych określonych na X może wyznaczać ten sam podzbiór A zbioru pełnego. Na przykład podzbiór $A = \{2, 4\}$ zbioru pełnego $X = \{1, 2, 3, 4, 5\}$ wyznaczony będzie przez funkcję zdaniową

$$P_1(x) = \mathrm{E}y\,(x = 2y),$$

jak i przez funkcję zdaniową

$$P_2(x) = [(x = 2) \vee (x = 4)].$$

Obie te funkcje mimo różnego zapisu wyznaczają tę samą własność przedmiotów ze zbioru pełnego X, mianowicie własność: być liczbą parzystą.

Spośród wszystkich funkcji zdaniowych dwa rodzaje zasługują na szczególną uwagę. Jeden rodzaj to funkcje zdaniowe $P(x)$ zawsze prawdziwe dla elementów zbioru pełnego (np. funkcja $x = x$), drugi — to ich zaprzeczenie, funkcje zdaniowe $\sim P(x)$ zawsze fałszywe dla elementów zbioru pełnego (np. funkcja $x \neq x$).

Zbiór przedmiotów spełniających każdą funkcję pierwszego rodzaju, będzie się składał ze wszystkich przedmiotów zbioru pełnego. Dla odróżnienia od zbioru pełnego zbiór ten traktowany jako podzbiór zbioru pełnego X, będziemy oznaczali przez 1. Jest więc

$$(2) \qquad \mathrm{A}x\,(x \epsilon 1).$$

Dla funkcji zdaniowych drugiego rodzaju, nie będzie przedmiotów które ją spełniają. Wyznaczony przez nie podzbiór zbioru pełnego oznaczamy symbolem 0 i nazywamy *podzbiorem pustym*. Żaden element nie należy do zbioru pustego. Jest więc

$$(3) \qquad \mathrm{A}x \sim (x \epsilon 0).$$

Gdy zbiór pełny jest ustalony, to jego podzbiory będziemy krótko nazywali *zbiorami*. Analogicznie mówić będziemy o zbiorze pełnym i zbiorze pustym, dla oznaczania podzbiorów 1 oraz 0.

Streszczenie

Określiliśmy pojęcie zbioru pełnego i podzbiory tego zbioru jako własności elementów zbioru pełnego. Każda funkcja zdaniowa wyznacza pewien podzbiór zbioru pełnego. Zdefiniowaliśmy podzbiór pełny i podzbiór pusty.

Zadania

1. Określić zbiory leżące na prostej liczbowej (tzn. podzbiory zbioru pełnego składającego się z liczb rzeczywistych) odpowiadające następującym funkcjom zdaniowym:

$$x < 0, \quad 2 < x \quad \text{i} \quad x \leqslant 5, \quad x = 7, \quad x^2 + 5x + 6 = 0,$$
$$x + 1 = x, \quad x = x, \quad x = x^2.$$

2. Niech zbiór pełny X będzie zbiorem liczb naturalnych. Przy użyciu relacji $x = y$, $z = x + y$, $z = x \cdot y$, $x | y$, (x dzieli y) i kwantyfikatorów zapisać funkcje zdaniowe jednej zmiennej wyznaczające zbiory:
a) liczb parzystych; b) liczb nieparzystych; c) liczb pierwszych; d) zbiór złożony z jednej liczby 3; e) zbiór pełny 1; f) zbiór pusty 0.

§ 29. DZIAŁANIA NA ZBIORACH

Obierzmy jakiś zbiór pełny X przedmiotów. Weźmy dwa zbiory A i B (tzn. podzbiory zbioru X) i określmy nowy zbiór złożony z tych elementów zbioru pełnego X które należą do A *lub* B. Precyzyjniej możemy zbiór C określić następująco:

$$\mathrm{A}x\,[(x \epsilon C) \equiv (x \epsilon A) \vee (x \epsilon B)]$$

Zbiór ten nazywamy *sumą zbiorów* A i B i oznaczamy przez $A \cup B$. Znak „$\cup$" jest znakiem działania, które dwóm zbiorom A i B przyporządkowuje nowy zbiór $C = A \cup B$. Na przykład jeżeli $A = \{1, 3, 5, 6, 9\}$, zaś $B = \{2, 3, 5, 6\}$, to $A \cup B = \{1, 2, 3, 5, 6, 9\}$.

W podobny sposób określamy *przekrój zbiorów A i B*. Jest to zbiór złożony z tych elementów zbioru pełnego, które należą jednocześnie i do zbioru A i do zbioru B, tj.

$$\mathrm{A}x[(x \in D) \equiv (x \in A) \,\&\, (x \in B)].$$

Przekrój zbiorów A i B oznaczamy symbolem $A \cap B$.

Przekrojem zbiorów $A = \{1, 3, 5, 6, 9\}$ i $B = \{2, 3, 5, 6\}$ jest $A \cap B = \{3, 5, 6\}$.

Przyjęliśmy więc następującą definicję sumy i przekroju zbiorów:

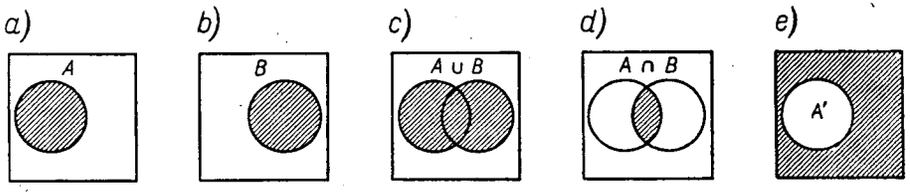
DEFINICJA 1. *Sumą $A \cup B$ i przekrojem $A \cap B$* zbiorów A oraz B nazywamy takie zbiory, dla których

$$\mathrm{A}x[(x \in A \cup B) \equiv (x \in A) \vee (x \in B)],$$

$$\mathrm{A}x[(x \in A \cap B) \equiv (x \in A) \,\&\, (x \in B)].$$

Słownie naszą definicję moglibyśmy wypowiedzieć następująco. Elementy zbioru A mają pewną własność P. Elementy zbioru B — pewną własność Q. Elementy sumy $A \cup B$ mają własność $P \vee Q$ (przynajmniej jedną z własności P, Q), elementy zaś przekroju mają własność $P \,\&\, Q$ (obie te własności łącznie).

Suma i przekrój zbiorów mają prostą interpretację geometryczną, podaną na rysunku 1. Zbiór pełny jest przedstawiony na tym rysunku



Rys. 1. Interpretacja sumy i przekroju zbiorów

jako kwadrat. Jego elementami są punkty. Elementy x zbioru pełnego takie, że $x \in A$ leżą wewnątrz zakreskowanego okręgu (rys. 1a). Podobnie rysunek 1b przedstawia elementy x takie, że $x \in B$. Na rysunku 1c wyróżniono takie elementy x, które spełniają funkcję zdaniową $(x \in A) \vee (x \in B)$, czyli takie, które należą do zbioru $A \cup B$. Podobnie na rysunku 1d wyróżniono takie x, które spełniają funkcję zdaniową $(x \in A) \,\&\, (x \in B)$, czyli te, które należą do $A \cap B$.

Podamy teraz definicję dopełnienia zbioru. *Dopełnienie zbioru A*, to zbiór tych elementów zbioru pełnego X, które do zbioru A nie należą. Dopełnienie zbioru A oznaczamy symbolem A'. Na rysunku 1e wyróżniono te x, które należą do zbioru pełnego i spełniają funkcję zdaniową „x nie należy do A", czyli x należące do dopełnienia A' zbioru A.

DEFINICJA 2. *Dopełnieniem zbioru A* nazywamy taki zbiór A', że:

$$\mathrm{A}x[(x \in A') \equiv \sim(x \in A)].$$

Jeżeli zbiór pełny był zbiorem liczb $\{1, 2, \ldots, 10\}$, to dopełnieniem zbioru $A = \{1, 3, 5, 6, 9\}$ będzie zbiór $A' = \{2, 4, 7, 8, 10\}$.

Z tego co powiedzieliśmy na początku tego rozdziału wynika następująca definicja równości:

DEFINICJA 3. Dwa *zbiory* A i B są *równe* wtedy i tylko wtedy, gdy

$$\mathrm{A}x[(x \in A) \equiv (x \in B)]. \tag{1}$$

Określmy jeszcze *stosunek zawierania* (albo inaczej inkluzji) między zbiorami. Będziemy mówić, że zbiór A jest zawarty w B wtedy i tylko wtedy, gdy każdy element zbioru A jest elementem zbioru B. Stosunek zawierania zbiorów oznaczamy zazwyczaj symbolem „$\subset$" i nazywamy *inkluzją*. Jeżeli chcemy napisać, że zbiór A jest zawarty w B piszemy krótko $A \subset B$.

DEFINICJA 4. Mówimy, że między zbiorami A i B zachodzi *stosunek inkluzji*, co zapisujemy $A \subset B$, wtedy i tylko wtedy, gdy

$$\mathrm{A}x[(x \in A) \Rightarrow (x \in B)].$$

Każdy zbiór jest zawarty w zbiorze 1:

$$A \subset 1. \tag{2}$$

Rzeczywiście trzeba sprawdzić, że

$$\mathrm{A}x[(x \in X) \Rightarrow (x \in 1)].$$

Ponieważ jednak mieliśmy $\mathrm{A}x(x \in 1)$, więc następnik implikacji stojącej w wyrażeniu (1) pod kwantyfikatorem będzie prawdziwy, gdy x będzie

dowolnym konkretnym elementem zbioru X. A więc prawdziwa będzie implikacja

$$(x \in A) \Rightarrow (x \in 1)$$

dla dowolnego elementu $x \in X$, czyli (2) będzie prawdziwe.

Podobnie łatwo można sprawdzić, że każdy zbiór jest zawarty w sobie samym

(3) $$A \subset A.$$

Rzeczywiście wyrażenie $\mathrm{A}x[(x \in A) \Rightarrow (x \in A)]$ jest prawdziwe, gdyż daje się uzyskać z tautologii $p \Rightarrow p$, przez podstawienie za p funkcji zdaniowej $x \in A$, i opatrzenia wyrażenia dużym kwantyfikatorem.

Dla inkluzji „$\subset$" zachodzi następujące prawo

(4) $(A \subset B)\ \&\ (B \subset A)$ wtedy i tylko wtedy, gdy $A = B$.

Fakt ten stanowi po prostu inny zapis definicji równości zbiorów.

Podamy jeszcze *prawo przechodniości inkluzji zbiorów*:

(5) Jeżeli $A \subset B$ i $B \subset C$, to $A \subset C$.

Streszczenie

Zdefiniowaliśmy sumę $A \cup B$, iloczyn $A \cap B$ i uzupełnienie A' zbiorów. Określiliśmy zawieranie (inkluzję) między zbiorami A, B, i podaliśmy, że jest ona zwrotna i przechodnia, tzn. że spełnia prawa (3) i (5). Równość zbiorów $A = B$ podana w definicji 3 może być także określona prawem (4).

Zadania

1. Wychodząc z tautologii rachunku zdań

$$(p \Rightarrow q)\ \&\ (q \Rightarrow r) \Rightarrow (p \Rightarrow r),$$

udowodnić prawo przechodniości inkluzji zbiorów.

2. Udowodnić, że zbiór pusty 0 zawarty jest w każdym zbiorze:

$$0 \subset A.$$

3. a) Udowodnić, że jeżeli $A \subset B$, to $B' \subset A'$.

b) Udowodnić, że jeżeli $A \subset B$, to $A \cup C \subset B \cup C$ i $A \cap C \subset B \cup C$.

c) Wywnioskować stąd następujące prawa:

jeżeli $A = B$, to $A \cup C = B \cup C$;

jeżeli $A = B$, to $A \cap C = B \cap C$;

jeżeli $A = B$, to $A' = B'$.

4. Opierając się tylko na zadaniu 2 i prawach dotyczących inkluzji podanych w tekście tego paragrafu udowodnić, że równość jest

a) zwrotna: $A = A$;

b) przechodnia: jeżeli $A = B$ i $B = C$, to $A = C$;

c) symetryczna: jeżeli $A = B$, to $B = A$.

5. a) Niech zbiór pełny X będzie zbiorem liczb rzeczywistych (punktów prostej liczbowej), A zaś — przedziałem $1 \leqslant x \leqslant 5$, B przedziałem $3 \leqslant x \leqslant 6$. Znaleźć $A \cup B$, $A \cap B$, A' i B'.

b) Znaleźć $A \cup B$, $A \cap B$, A' i B', gdy A jest półprostą $x < 0$, B półprostą $x \geqslant 0$.

6. Udowodnić, że z definicji uzupełnienia zbiorów i równości zbiorów wynika, że $1' = 0$ oraz $0' = 1$.

§ 30. PRAWA RACHUNKU ZBIORÓW

Dzięki odpowiedniości między działaniami na zbiorach a spójnikami zdaniowymi można bardzo łatwo podać i udowodnić cały szereg praw rachunku zbiorów.

W paragrafie tym pokażemy jak się takie prawa znajduje i dowodzi. Na początek udowodnimy następujące prawo:

(1) $$A \cup A = A, \quad A \cap A = A.$$

Wychodząc z tautologii rachunku zdań

$$p \vee p \equiv p, \text{ względnie } p\ \&\ p \equiv p,$$

mamy, podstawiając za p funkcję zdaniową $x \in A$,

$$(x \in A) \vee (x \in A) \equiv (x \in A),$$

$$(x \in A)\ \&\ (x \in A) \equiv (x \in A).$$

Stąd opatrując równoważność dużym kwantyfikatorem, otrzymujemy

$$\mathrm{A}x[(x\epsilon A) \vee (x\epsilon A) \equiv (x\epsilon A)];$$

względnie

$$\mathrm{A}x[(x\epsilon A) \,\&\, (x\epsilon A) \equiv (x\epsilon A)].$$

Zgodnie z definicją sumy i przekroju zbiorów daje to

$$\mathrm{A}x[(x\epsilon A \cup A) \equiv (x\epsilon A)],$$

względnie

$$\mathrm{A}x[(x\epsilon A \cap A) \equiv (x\epsilon A)].$$

Stąd z definicji równości zbiorów otrzymujemy właśnie prawa (1). Prawa te są bardzo ciekawe. Mówią one, że sumowanie zbiorów i przekrój zwany również *iloczynem zbiorów* są *operacjami idempotentnymi* tzn. wykonane na takich samych zbiorach dowolną ilość razy dają w wyniku pierwotny zbiór.

Pokażemy teraz przykładowo jak z tautologii rachunku zdań można wyprowadzić odpowiadające jej prawo rachunku zbiorów. Weźmy tautologię zwaną prawem podwójnego przeczenia:

$$\sim(\sim p) \equiv p.$$

Stosując podstawienie takie jak poprzednio za p, otrzymamy

$$\mathrm{A}x[\sim(\sim(x\epsilon A)) \equiv (x\epsilon A)],$$

czyli

$$\mathrm{A}x[\sim(x\epsilon(A')) \equiv (x\epsilon A)],$$

$$\mathrm{A}x[(x\epsilon(A'))' \equiv (x\epsilon A)].$$

Z definicji równości zbiorów otrzymamy wyprowadzone z prawa podwójnego przeczenia prawo rachunku zbiorów.

(2) $$(A')' = A$$

mówiące, że operacja dwukrotnego dopełniania zbiorów jest operacją idempotentną.

W podobny sposób możemy otrzymać prawa de Morgana dla rachunku zbiorów; pierwsze z nich to prawo

(3) $$(A \cup B)' = A' \cap B'.$$

Z pierwszego prawa de Morgana dla rachunku zdań

$$\sim(p \vee q) \equiv \sim p \,\&\, \sim q,$$

podstawiając za p funkcję zdaniową $x\epsilon A$, za q zaś funkcję zdaniową $x\epsilon B$, otrzymamy

$$\sim[(x\epsilon A) \vee (x\epsilon B)] \equiv \sim(x\epsilon A) \,\&\, \sim(x\epsilon B).$$

Stąd mamy

$$\mathrm{A}x\big[\sim\big((x\epsilon A) \vee (x\epsilon B)\big) \equiv \sim(x\epsilon A) \,\&\, \sim(x\epsilon B)\big],$$

czyli

$$\mathrm{A}x[\sim(x\epsilon A \cup B) \equiv (x\epsilon A') \,\&\, (x\epsilon B')],$$

dalej

$$\mathrm{A}x\big[\big(x\epsilon(A \cup B)'\big) \equiv \big(x\epsilon(A' \cap B')\big)\big],$$

w myśl definicji 2 daje to prawo:

$$(A \cup B)' = A' \cap B'.$$

Podobnie z tautologii stanowiącej drugie prawo de Morgana dla rachunku zdań:

$$\sim(p \,\&\, q) \equiv \sim p \vee \sim q,$$

możemy otrzymać drugie prawo de Morgana dla rachunku zbiorów:

(4) $$(A \cap B)' = A' \cup B'.$$

Z prawa wyłączonego środka

$$p \vee \sim p,$$

podstawiając za p funkcję zdaniową $x\epsilon A$, gdzie A jest dowolnym zbiorem, otrzymamy

$$(x\epsilon A) \vee \sim(x\epsilon A),$$

czyli

$$\mathrm{A}x[(x\epsilon A) \vee \sim(x\epsilon A)],$$

więc

$$\mathrm{A}x[(x\epsilon A) \vee (x\epsilon A')],$$

co daje

$$\mathrm{A}x[x\epsilon(A \cup A')].$$

Wobec definicji zbioru pełnego i definicji równości zbiorów otrzymujemy prawo

(5) $$A \cup A' = 1.$$

mówiące, że suma zbioru i jego uzupełnienie daje zbiór pełny X.

Podobnie z prawa wyłączonej sprzeczności

$$\sim(p \,\&\sim p)$$

otrzymujemy

$$\mathrm{A}x[\sim((x \epsilon A) \,\&\sim(x \epsilon A))],$$

czyli

$$\mathrm{A}x[\sim(x \epsilon (A \cap A'))],$$

co wobec

$$\mathrm{A}x[\sim(x \epsilon 0)]$$

i definicji zbioru pustego daje prawo

(6) $$A \cap A' = 0$$

mówiąc, że przekrój każdego zbioru ze swoim uzupełnieniem jest zbiorem pustym.

Zadania

1. Wychodząc z tautologii $p \vee q = q \vee p$ oraz $p \,\& q \equiv q \,\& p$, udowodnić prawa rachunku zbiorów

$$A \cup B = B \cup A \quad \text{i} \quad A \cap B = B \cap A.$$

2. Z jakich tautologii możemy otrzymać następujące prawa:

$$A \cup (B \cup C) = (A \cup B) \cup C,$$

$$A \cap (B \cap C) = (A \cap B) \cap C?$$

3. Jakie prawa otrzymamy, wychodząc z tautologii

$$p \,\&\, (q \vee r) \equiv p \,\& q \vee p \,\& r$$

oraz

$$p \vee (q \,\& r) \equiv (p \vee q) \,\&\, (p \vee r)?$$

4. a) Udowodnić drugie prawo de Morgana (prawo (4)).
b) Zinterpretować oba prawa na rysunku takim jak rys. 1.

5. Określmy operację różnicy $A - B$ dwóch zbiorów A i B następująco:

$$\mathrm{A}x[(x \epsilon A{-}B) \equiv x \epsilon A \,\& \sim (x \epsilon B)].$$

a) Podać interpretację operacji różnicy na wykresie analogicznym do rysunku 1.
b) Udowodnić, że

$$A \subset (A{-}B) \cup B$$

i podać przykład, że może być

$$A \neq (A{-}B) \cup B.$$

Wskazać tautologie rachunku zdań z jakich należy korzystać w dowodzie.
c) Udowodnić prawo

$$(A{-}B){-}C = A{-}(B \cup C).$$

d) Udowodnić prawo

$$(A{-}B) \cap (A{-}C) = A{-}(B \cup C).$$

e) Czemu się równa: $1{-}A$, $A{-}1$, $A{-}0$, $0{-}A$?

6. Operacja $A \div B = (A{-}B) \cup (B{-}A)$ nazywa się *różnicą symetryczną.*
a) Podać interpretację operacji różnicy symetrycznej na rysunku analogicznym do rysunku 1.
b) Udowodnić, że $A \div A = 0$.
c) Udowodnić, że $(A \div B) \div C = A \div (B \div C)$ oraz $A \div B = B \div A$.
d) Udowodnić, że $0 \div A = A$ oraz $1 \div A = A'$.
e) Udowodnić, że $A \cap (B \div C) = (A \cap B) \div (A \cap C)$.
f) Udowodnić, że jeżeli $A \cap B = 0$, to $A \div B = A \cup B$.

7. a) Udowodnić, że $A \subset B$ wtedy i tylko wtedy, gdy $A \cup B = B$.
b) Udowodnić, że $A \subset B$ wtedy i tylko wtedy, gdy $A \cap B = A$.
c) Udowodnić, że $A \subset B$ wtedy i tylko wtedy, gdy $B \cup A' = 1$.
d) Udowodnić, że $A \subset B$ wtedy i tylko wtedy, gdy $B' \cap A = 0$.

§ 31. KOMBINATORYKA

Nie roszcząc sobie pretensji do ścisłości tego sformułowania, możemy powiedzieć, że kombinatoryka bada liczbę elementów zbiorów skończonych. Jako taka może być uznawana za część fragmentu teorii mnogości, zajmującego się zbiorami skończonymi.

Kombinatoryka nie jest więc działem związanym bezpośrednio z rachunkiem zbiorów. Podaje ona jednak wzory pozwalające obliczyć np. liczbę podzbiorów zbioru skończonego, liczbę podzbiorów zawierających

po k elementów, liczbę funkcji, które mają argumenty w jednym zbiorze skończonym a wartości w drugim zbiorze skończonym.

Podamy tylko najważniejsze wzory z kombinatoryki dotyczące zbiorów, ze względu na praktyczne zastosowania. Wzory dotyczące funkcji znajdzie czytelnik w ostatnim paragrafie następnego rozdziału dotyczącego funkcji.

TWIERDZENIE 1. *Jeżeli zbiór pełny X jest skończony i ma n elementów, to zbiór wszystkich jego podzbiorów ma 2^n elementów.*

Dowód. Jeżeli $n=0$, tzn. gdy zbiór X jest pusty, to ma on tylko jeden podzbiór — mianowicie pusty. Teza twierdzenia jest dla $n=0$ prawdziwa, gdyż ilość podzbiorów wynosi $2^0=1$.

Dalej dowód będziemy prowadzić przez indukcję względem n. W tym celu winniśmy udowodnić, że dla dowolnego s jeśli zbiór s-elementowy ma 2^s podzbiorów, to zbiór $(s+1)$-elementowy ma 2^{s+1} podzbiorów.

Niech zbiór s-elementowy ma elementy $a_1, a_2, \ldots, a_s$, zbiór zaś $(s+1)$-elementowy te same elementy oraz jeszcze element a_{s+1}, a więc elementami tego zbioru będą $a_1, \ldots, a_{s+1}$. Wypiszemy wszystkie 2^s podzbiorów $A, B, C, D, \ldots$ zbioru $\{a_1, \ldots, a_s\}$, s-elementowego. Będą to niektóre podzbiory zbioru $\{a_1, \ldots, a_s, a_{s+1}\}$ $(s+1)$-elementowego. Dopisując do każdego z podzbiorów

$$A,\ B,\ C,\ D,\ \ldots \tag{1}$$

jeszcze podzbiory otrzymane z poprzednich przez dołączenie do każdego z nich elementu a_{s+1}, otrzymamy podzbiory

$$\{A, a_{s+1}\},\quad \{B, a_{s+1}\},\quad \{C, a_{s+1}\},\quad \{D, a_{s+1}\},\ \ldots \tag{2}$$

Wszystkie podzbiory (1) i (2) będą różne (dlaczego?) i łącznie będą dawać wszystkie podzbiory zbioru.

W wierszu (1) jak i w wierszu (2) stoi po 2^s podzbiorów. Razem więc podzbiorów zbioru $(s+1)$-elementowego jest

$$2^s+2^s=2\cdot 2^s=2^{s+1}.$$

Na mocy zasady indukcji twierdzenie 1 zostało udowodnione dla każdej liczby naturalnej n.

Zajmiemy się teraz wyznaczeniem liczby tych podzbiorów zbioru pełnego X, n-elementowego, które mają k elementów. Oznaczmy liczbę tych podzbiorów przez $\binom{n}{k}$.

Oczywiście jest tylko jeden podzbiór zbioru X, zero-elementowy — podzbiór pusty. A więc $\binom{n}{0}=1$.

Udowodnimy teraz, że dla dowolnego k, $0\leqslant k\leqslant n-1$

$$\binom{n}{k+1}=\binom{n}{k}\cdot\frac{n-k}{k+1}. \tag{3}$$

Jeżeli zbiór X ma n elementów $a_1, a_2, \ldots, a_n$, A zaś jest jakimś podzbiorem k-elementowym, np. dla ustalenia uwagi podzbiorem $\{a_1, \ldots, a_k\}=A$, to dołączając do A którykolwiek z pozostałych $n-k$ elementów $a_{k+1}, \ldots, a_n$ otrzymamy z podzbioru A, $n-k$ podzbiorów $(k+1)$-elementowych $\{a_1, a_2, \ldots, a_k, a_{k+1}\}=\{A, a_{k+1}\}$, $\{a_1, \ldots, a_k, a_{k+2}\}=\{A, a_{k+2}\}$, $\ldots$, $\{a_1, \ldots, a_k, a_n\}=\{A, a_n\}$. Niech $A, B, C, \ldots$ będą wszystkimi podzbiorami X k-elementowymi. Przez opisaną operację otrzymamy z nich $\binom{n}{k}$. $(n-k)$ zbiorów $(k+1)$-elementowych. Nie wszystkie one będą różne. Na przykład ze zbiorów: $A_1=\{a_1, \ldots, a_k\}$, $A_2=\{a_1, \ldots, a_{k-1}, a_{k+1}\}$, $\ldots$, $A_{k+1}=\{a_2, \ldots, a_k, a_{k+1}\}$, dołączając odpowiednie elementy $a_{k+1}, a_k, \ldots, a_1$ otrzymamy $k+1$ identycznych podzbiorów $\{a_1, \ldots, a_k, a_{k+1}\}$. A więc pośród $\binom{n}{k}\cdot(n-k)$ otrzymanych zbiorów, będą wszystkie podzbiory $(k-1)$-elementowe i każdy z nich będzie występował $k+1$ razy. A więc $\binom{n}{k+1}=\dfrac{\binom{n}{k}\cdot(n-k)}{k+1}=\binom{n}{k}\cdot\dfrac{n-k}{k+1}$, czyli zgodnie z wzorem (3).

Ze wzoru (3) otrzymujemy przez indukcję natychmiast twierdzenie:

TWIERDZENIE 2. *Liczba podzbiorów k-elementowych zbioru n-elementowego oznaczana przez $\binom{n}{k}$, wynosi*

$$\binom{n}{k}=\frac{n\cdot(n-1)\ldots(n-k+1)}{1\cdot 2\cdot\ldots\cdot k},\qquad \text{dla } k=1, \ldots, n;$$

$\binom{n}{0}=1$, $\binom{n}{k}=0$, *zaś dla* $k>n$.

Wzór ten jest bardzo łatwy do zapamiętania. W liczniku i mianowniku jest k czynników. W liczniku maleją one o 1 zaczynając od n, w mianowniku rosną o 1, zaczynając od 1.

Na przykład będziemy mieli $\binom{n}{1} = n$ podzbiorów jednoelementowych zbioru n-elementowego, $\binom{n}{2} = \frac{n(n-1)}{2}$ podzbiorów dwuelementowych zbioru n-elementowego, itp. Ale na przykład zbiór pięcioelementowy nie będzie miał wcale podzbiorów siedmioelementowych, więc $\binom{5}{7} = 0$.

Streszczenie

Udowodniliśmy, że liczba podzbiorów zbioru n-elementowego wynosi 2^n. Liczba $\binom{n}{k}$ podzbiorów k-elementowych zbioru n-elementowego, dana jest za pomocą wzoru

$$\binom{n}{k} = \frac{n(n-1)\dots(n-k+1)}{1\cdot 2\cdot\dots\cdot k} \quad \text{dla } k = 1, 2, \dots, n,$$

$$\binom{n}{0} = 1, \text{ zaś } \binom{n}{k} = 0 \text{ dla } k > n.$$

Zadania

1. Wypisać wszystkie podzbiory zbioru trójelementowego $\{a_1, a_2, a_3\}$. Potem wypisać wszystkie zbiory uzyskane przez dopisanie do każdego ze zbiorów trójelementowych elementu a_1, tak jak w dowodzie twierdzenia 2. W której grupie będzie zbiór pusty 0, a w której grupie zbiór 1, zbioru pełnego $\{a_1, a_2, a_3, a_4\}$

2. Udowodnić z definicji $\binom{n}{k}$, że $\binom{n}{k} = \binom{n}{n-k}$.

3. Udowodnić, że $\binom{n}{k} + \binom{n}{k+1} = \binom{n+1}{k+1}$, również wychodząc z podanej w książce definicji.

4. Z definicji $\binom{n}{k}$ oraz z twierdzenia 1 wywnioskować, że

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{n} = 2^n.$$

Zinterpretować ten wzór.

5. Udowodnić wzór Newtona:

$$(a+b)^n = \binom{n}{0} a^n + \binom{n}{1} a^{n-1} b + \binom{n}{2} a^{n-2} b^2 + \dots + \binom{n}{n} b^n$$

obliczając $(a+b_1)\cdot(a+b_2)(a+b_3)\dots(a+b_n)$ i kładąc $b_1 = b_2 = \dots = b_n = b$.

6. Udowodnić, że

$$\binom{n}{0} - \binom{n}{1} + \binom{n}{2} - \binom{n}{3} + \dots + (-1)^n \binom{n}{n} = 0.$$

Czy można uzasadnić ten wzór nie korzystając z wzoru Newtona?

7. Zapisać wzór (4) w postaci

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

Rozdział 6

RELACJE

W rozdziale tym omówimy pojęcie relacji. W szczególności szerzej zajmiemy się relacjami równoważności (§ 33) i omówimy znaczenie dla matematyki tak zwanej zasady abstrakcji (§ 34). W dalszym ciągu omówimy pojęcie relacji częściowego porządku (§ 35). W paragrafie przedostatnim podamy definicję struktury (§ 36). Paragraf ostatni (§ 37) poświęcimy funkcjom.

Sądzimy, że znajomość materiału zawartego w tym rozdziale ułatwi czytelnikowi zrozumienie wielu rozważań podanych w tej książce. Ułatwi również lekturę prac specjalistycznych, czy to z matematyki, czy logiki, względnie bardziej specjalnych prac z teorii automatów, czy też lingwistyki matematycznej. Materiał wykładany w tym rozdziale można znaleźć w książkach Mostowskiego, Słupeckiego i Borkowskiego, Rasiowej [1969].

§ 32. DEFINICJA RELACJI

Definiując relację wychodzi się od pojęcia pary uporządkowanej. *Parą uporządkowaną* (x, y) dwóch elementów nazywamy taką parę, w której gra rolę porządek elementów. Element x jest pierwszym elementem pary, a element y drugim. Tak więc dla par uporządkowanych pary (x, y) i (y, x) będą różne, chyba że x jest równe y.

Zbiór Z, którego elementami są wszystkie uporządkowane pary x, y, gdzie $x \in X$, zaś $y \in Y$, nazywamy *produktem kartezjańskim* i oznaczamy $Z = X \times Y$. Na przykład jeżeli zbiór X będzie się składał z dwóch ludzi: Jana i Pawła, zbiór Y zaś z pięciu potraw: krupniku, befsztyku, lodów, budyniu, ciastek, to produkt kartezjański będzie się składał z dziesięciu par:

(Jan, krupnik), (Paweł, krupnik),
(Jan, befsztyk), (Paweł, befsztyk),
(Jan, lody), (Paweł, lody),
(Jan, budyń), (Paweł, budyń),
(Jan, ciastka), (Paweł, ciastka).

Elementy ze zbioru X (ludzie) stoją zawsze na pierwszym miejscu pary, elementy zaś zbioru Y (potrawy) na drugim.

Jeżeli weźmiemy zbiór X złożony z liczb od jeden do sześciu

$$X = \{1, 2, 3, 4, 5, 6\},$$

to produkt kartezjański $X \times X$ składać się będzie z 36 par uporządkowanych (i, j) dla wszystkich $1 \leqslant i \leqslant 6$, $1 \leqslant j \leqslant 6$.

Rozpatrzmy podzbiór U produktu $X \times X$ złożony z elementów (par uporządkowanych):

(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1, 6),
(2, 2), (2, 4), (2, 6), (3, 3), (3, 6),
(4, 4), (5, 5), (6, 6),

Jaka własność zachodzi między elementami x i y zbioru X takimi, że $(x, y) \in U$? Łatwo się przekonać, że dla dowolnych $x, y \in X$ para $(x, y) \in U$ wtedy i tylko wtedy, gdy x dzieli y, co zapisujemy:

$$\text{dla każdych } x, y \in X: \quad (x, y) \in U \equiv x|y.$$

Związek podzielności, czyli jak matematycy mówią, *relacja podzielności* liczb zbioru X została opisana przez podanie par tych elementów, które ją spełniają. Taki sposób określania relacji może się wydawać dziwnym, jednak w wielu przypadkach określenie relacji przez podanie zbioru wszystkich par tych elementów, które pozostają w relacji, jest jedynym sposobem jej określenia. Na przykład jeżeli chcemy określić relację: x lubi potrawę y, to musimy wyliczyć: Jan lubi krupnik, Jan lubi befsztyk, Paweł lubi lody, Paweł lubi budyń, Paweł lubi ciastka. Otrzymujemy wtedy opis pewnej relacji między dwoma ludźmi, a pięcioma potrawami.

Powyższe rozważania dają podstawę do następującej definicji relacji:

DEFINICJA 1. Zbiór R złożony z pewnych par uporządkowanych (x, y), gdzie x i y należą do X, tzn. podzbiór produktu kartezjańskiego $X \times X$,

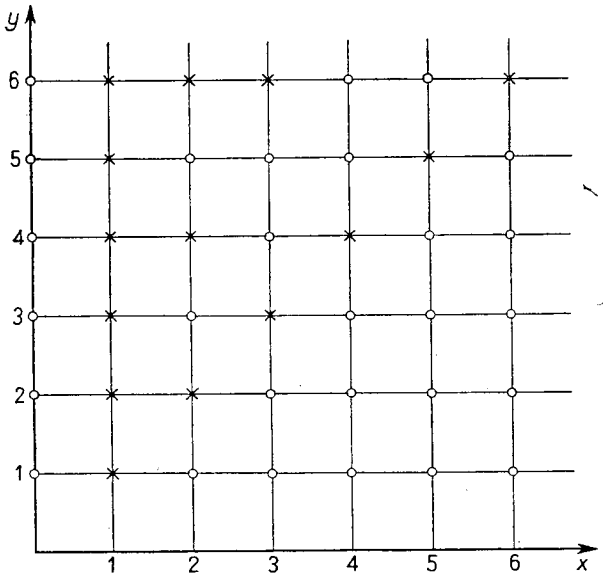
nazywać będziemy *relacją* określoną w zbiorze X (inaczej mówiąc *relacją dwuczłonową* o polu X).

Fakt, że para $(x, y) \in R$, zapisujemy następująco:

$$xRy \tag{1}$$

i czytamy: przedmiot x jest w relacji R z przedmiotem y.

Taka definicja relacji jest w istocie definicją geometryczną. Na ogół mówiąc o relacji rozumie się przez nią jakiś związek (1), a nie zbiór R mó-



Rys. 2. Wykres relacji $x|y$ w zbiorze $X = \{1, 2, 3, 4, 5, 6\}$

wiący dla jakich x i y ten związek zachodzi. Sam zbiór par nazywa się wtedy *wykresem relacji* R.

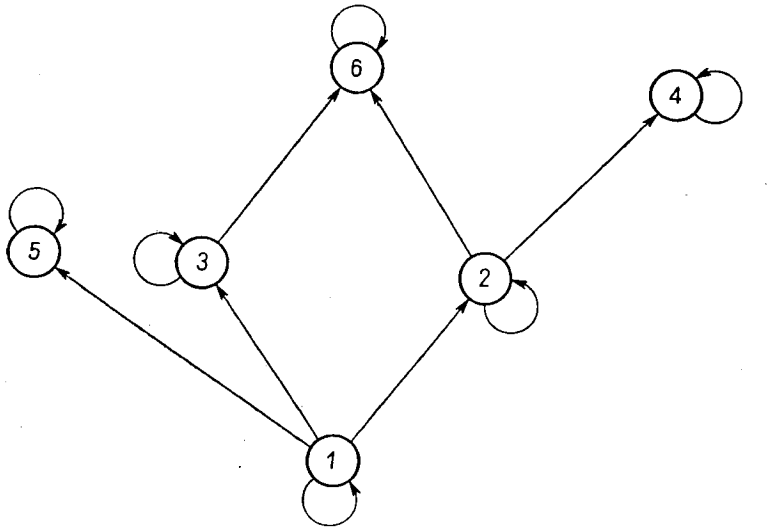
Przyjmując jednak definicję 1 jesteśmy w zgodzie z naturalnym rozumieniem tego, że jakiś związek jest określony wtedy, gdy wiemy dla jakich par on zachodzi, a dla jakich nie.

Relacje można bardzo prosto przedstawić geometrycznie Na przykład dla relacji określonej w przykładzie 1 zaznaczmy na osiach x i y punkty zbioru X. Parom (x, y) odpowiadać będą pewne punkty płaszczyzny. Na rysunku 2 pary ze zbioru R zaznaczone zostały krzyżykami.

Możliwe jest również przedstawienie relacji w postaci tzw. grafu tak jak na rysunku 3. Wypisaliśmy na nim elementy zbioru X połączone strzałkami. Relacja xRy zachodzi wówczas, gdy z punktu x możemy przejść do punktu y poruszając się zgodnie z kierunkami strzałek.

Podobnie określa się relację zachodzącą między przedmiotami dwóch różnych zbiorów.

Relacją zachodzącą między przedmiotami ze zbioru X a przedmiotami ze zbioru Y nazywać będziemy każdy podzbiór R produktu $X \times Y$. Fakt,



Rys. 3. Graf relacji $x|y$ (x dzieli y) w zbiorze $X = \{1, 2, 3, 4, 5, 6\}$

że para $(x, y) \in R$, zapisujemy xRy i czytamy: przedmiot x jest w relacji R do przedmiotu y. Zbiór X nazywamy w tym przypadku *dziedziną*, zbiór Y zaś *przeciwdziedziną relacji* R.

Na przykład, jeżeli podzbiór będzie się składał z par:

(Jan, krupnik), (Jan, befsztyk), (Paweł, lody),

(Paweł, budyń), (Paweł, ciastka), (Paweł, befsztyk),

to otrzymamy rozpatrywaną już relację „lubi" mającą zachodzić między dwojgiem ludzi, dziedziną relacji (zbiór X) i pięcioma potrawami, przeciwdziedziną relacji (zbiór Y).

Streszczenie

Podaliśmy definicję relacji dwuargumentowej określonej na zbiorze X. Każda relacja R jest pewnym podzbiorem $U \subset X \times X$. Podobnie określiliśmy relację o dziedzinie X i przeciwdziedzinie Y jako podzbiór $X \times Y$. Para (x, y) pozostaje w relacji R, co zapisujemy $x\mathrm{R}y$, wtedy i tylko wtedy, gdy $(x, y) \in \mathrm{R}$.

Zadania

1. Narysować wykresy relacji $x = y$, $x < y$ i $x^2 + y^2 = 1$ w zbiorze X liczb rzeczywistych.

2. Udowodnić, że wszystkich relacji określonych na zbiorze X, mającym n elementów jest 2^{n^2}. Wypisać tabelki wszystkich relacji określonych na zbiorze dwuelementowym.

3. Niech R i S będą dwoma relacjami w zbiorze X. Relację T taką, że

$$x\mathrm{T}y \equiv (x, y) \in \mathrm{R} \cup \mathrm{S}$$

nazywamy *sumą relacji* R i S i piszemy $\mathrm{R} \cup \mathrm{S} = \mathrm{T}$. Relację Q taką, że

$$x\mathrm{Q}y \equiv (x, y) \in \mathrm{R} \cap \mathrm{S}$$

nazywamy *iloczynem relacji* R i S i piszemy $\mathrm{R} \cap \mathrm{S} = \mathrm{Q}$.

Relację M taką, że

$$x\mathrm{M}y \equiv (x, y) \notin \mathrm{R}$$

nazywamy *negacją relacji* R i piszemy $\mathrm{M} = \mathrm{R}'$.

Udowodnić, że dla każdych $x, y \in X$:

$$x\mathrm{R} \cup \mathrm{S}y \equiv (x\mathrm{R}y \vee x\mathrm{S}y),$$

$$x\mathrm{R} \cap \mathrm{S}y \equiv (x\mathrm{R}y \,\&\, x\mathrm{S}y),$$

$$x\mathrm{R}'y \equiv \sim (x\mathrm{R}y).$$

Uwaga. Przy takim określeniu operacji „$\cup$", „$\cap$" i „$'$" zbiór wszystkich relacji określonych na zbiorze X tworzy algebrę Boole'a (por. rozdział VII). Jakimi relacjami są elementy wyróżnione 0 i 1 tej algebry?

4. Relację N taką, że:

$$x\mathrm{N}y \equiv \mathrm{E}z(x\mathrm{R}z \,\&\, z\mathrm{S}y)$$

nazywamy *iloczynem względnym* lub *złożeniem relacji* R i S i oznaczamy $\mathrm{S} \cdot \mathrm{R}$.

Pokazać na przykładzie, że

$$\mathrm{R} \cdot \mathrm{S} \neq \mathrm{S} \cdot \mathrm{R}.$$

Czy składanie relacji jest łączne, tzn. czy zachodzi równość

$$(\mathrm{R} \cdot \mathrm{S}) \cdot \mathrm{T} = \mathrm{R} \cdot (\mathrm{S} \cdot \mathrm{T})?$$

5. Podać definicję relacji trójargumentowej — ogólnie definicję relacji k-argumentowej. Relację taką zapisujemy zwykle $\mathrm{R}(x, y, z)$; tak jak zamiast $x\mathrm{R}y$ możemy pisać $\mathrm{R}(x, y)$. Udowodnić, że różnych relacji k-argumentowych na zbiorze X n-elementowych jest 2^{n^k}. Czy relacje dwuargumentowe można uważać za relacje trójargumentowe?

6. Udowodnić, że:

a) jeżeli $X_1 \subset X$, zaś $X_2 \subset Y$, to $X_1 \times X_2 \subset X \times Y$;

b) $(X \times Y) \cup (X \times Z) = X \times (Y \cup Z)$,
$(X \times Y) \cup (Z \times Y) = (X \cup Z) \times Y$;

c) $(X \times Y) \cap (X \times Z) = X \times (Y \cap Z)$,
$(X \times Y) \cap (Z \times Y) = (X \cap Z) \times Y$.

7. Uzasadnić, że liczba relacji, które mogą zachodzić między przedmiotami ze zbioru s-elementowego X, a przedmiotami zbioru r-elementowego Y jest równa $2^{r \cdot s}$.

8. a) Uzasadnić, że każda funkcja zdaniowa dwuargumentowa $\Phi(x, y)$ określona w zbiorze X wyznacza pewną relację R w zbiorze X, taką że $a\mathrm{R}b$, dla $a, b \in X$ zachodzi wtedy i tylko wtedy, gdy przedmioty $a, b \in X$ spełniają funkcję zdaniową $\Phi(x, y)$ (tzn. gdy $\Phi(a, b)$ jest zdaniem prawdziwym).

b) Funkcje zdaniowe $\Phi(x, y)$ oraz $\Psi(x, y)$ wyznaczają odpowiednio relację R i S. Napisać funkcje zdaniowe wyznaczające relacje $\mathrm{R} \cup \mathrm{S}$, $\mathrm{R} \cap \mathrm{S}$, R' oraz relację $\mathrm{R} \cdot \mathrm{S}$ (por. zad. 3 i 4).

§ 33. RELACJE RÓWNOWAŻNOŚCI

W tym paragrafie zajmiemy się szeroką klasą relacji zwanych relacjami równoważności, stanowiącą uogólnienie relacji równości. Omówimy więc najpierw relację równości. Relacja równości „$=$" zachodzi w zbiorze X między dowolnym przedmiotem a jedynie nim samym. Jest to relacja określona przez zbiór par E postaci (x, x). Ma ona szereg ważnych własności:

1. Dla każdego $x \in X$: $x = x$.
2. Dla każdych $x, y \in X$: jeżeli $x = y$, to $y = x$.

3. Dla każdych $x, y, z \in X$: jeżeli $x = y$ i $y = z$, to $x = z$. Własności te wyrażamy mówiąc, że relacja ta jest:

zwrotna (własność 1),

symetryczna (własność 2),

przechodnia (własność 3).

Ponadto cechą szczególną równości jest następująca własność

4. Dla każdych $x, y, z, t \in X$: jeżeli $x = z$ i $y = t$, to $x\mathrm{R}y \leftrightarrow z\mathrm{R}t$ dla dowolnej relacji R określonej na zbiorze X. Jest ona zwana *własnością ekstensjonalności.*

Podane cztery własności charakteryzują relację równości. Można łatwo udowodnić, że każda relacja określona w zbiorze X i spełniająca warunki 1-4 musi być identyczna z relacją równości „=".

Dalej zajmiemy się relacjami określonymi w jakimś zbiorze X i spełniającymi następujące trzy warunki będące odpowiednikiem własności 1-3:

5. Prawo zwrotności: dla każdego $x \in X$: $x\mathrm{R}x$.

6. Prawo symetrii: dla każdych $x, y \in X$: jeżeli $x\mathrm{R}y$, to $y\mathrm{R}x$.

7. Prawo przechodniości: dla każdych $x, y, z \in X$: jeżeli $x\mathrm{R}y$ i $y\mathrm{R}z$, to $x\mathrm{R}z$.

Wprowadzimy następującą definicję.

DEFINICJA 2. *Relację* R określoną na zbiorze X nazywamy *równoważnością* w tym zbiorze, jeżeli spełnia ona warunki 5, 6 i 7 (zwrotności, symetrii i przechodniości).

Jedną z relacji równoważności jest jak widzimy relacja równości. Na podanych dalej przykładach zobaczymy, że relacji takich jest więcej. Przekonamy się tym samym, że własności 1-3 bez prawa ekstensjonalności 4 nie charakteryzują równości.

Podamy teraz przykłady relacji równoważności.

Relacja przystawania figur geometrycznych jest relacją równoważności w zbiorze figur geometrycznych. Prawa zwrotności, symetrii i przechodniości uznawane są często po prostu za jedne z aksjomatów relacji przystawania figur geometrycznych.

Podobnie relacja równoległości prostych jest relacją równoważności w zbiorze wszystkich prostych na płaszczyźnie.

Natomiast relacja podzielności określona w zbiorze liczb naturalnych 1, 2, 3, ... jak również relacja podzielności omawiana w przykładzie 1, z poprzedniego paragrafu nie są równoważnościami. Wprawdzie prawo

5 zwrotności i prawo 7 przechodniości są spełnione, natomiast prawo 6 symetrii nie jest spełnione, gdyż np.: 2 dzieli 4, lecz 4 nie dzieli 2.

Dalsze przykłady relacji równoważności znajdują się w zadaniach.

Streszczenie

Relację R określoną w zbiorze X nazywamy w tym zbiorze równoważnością, jeżeli jest ona zwrotna, symetryczna i przechodnia (warunki 5-7). Relacje równoważności stanowią uogólnienie pojęcia równości.

Zadania

1. Udowodnić, że relacja I określona w zbiorze X, zachodząca dla dowolnych dwóch elementów z tego zbioru (tzw. *relacja pełna*) jest relacją równoważności . Udowodnić, że relacja zachodząca tylko między przedmiotem a nim samym jest relacją równoważności. Co to za relacja?

2. Niech X będzie zbiorem podzbiorów $X, Y, Z, \ldots$ ustalonego zbioru A. Określmy w zbiorze X *relację równoliczności* X rwl Y, następująco: X rwl Y wtedy i tylko wtedy, gdy istnieje odwzorowanie jednoznaczne zbioru X na zbiór Y. (Patrz ostatni paragraf tego rozdziału).

Udowodnić, że relacja równoliczności jest równością w zbiorze X.

3. a) Udowodnić, że relacja zachodząca pomiędzy liczbami zespolonymi z oraz t wtedy i tylko wtedy, gdy $|z| = |t|$, jest relacją równoważności.

b) Udowodnić, że relacja zachodząca między z oraz t wtedy i tylko wtedy, gdy istnieje takie c będące liczbą rzeczywistą $\neq 0$, że $z = ct$, jest relacją równoważności w zbiorze liczb zespolonych.

c) Udowodnić, że relacja zachodząca między z oraz t wtedy i tylko wtedy, gdy $z-t$ jest liczbą rzeczywistą, jest relacją równoważności.

Jaki jest sens geometryczny tych relacji?

4. Udowodnić, że relacja zachodząca pomiędzy liczbami naturalnymi x i y wtedy i tylko wtedy, gdy $x+y$ jest liczbą parzystą, jest relacją równoważności.

§ 34. ZASADA ABSTRAKCJI

W paragrafie tym zajmiemy się ważnym twierdzeniem dotyczącym relacji równoważności, tzw. *zasadą abstrakcji.* Rozpatrzmy wpierw następujący przykład.

PRZYKŁAD 2. Rozpatrzmy zbiór X wektorów związanych, leżących na płaszczyźnie, tzn. wektorów zaczepionych w jakichś punktach. Dwa takie wektory będziemy uważać, za równoważne, jeżeli jeden z nich można tak przesunąć równolegle by pokrył się z drugim. Tak określona relacja między wektorami, będzie zwrotna, symetryczna i przechodnia, więc będzie równoważnością nie tylko z nazwy, lecz również w sensie definicji 2. Wektory związane dzielimy na klasy, abstrahując od ich położenia (punktu zaczepienia), zaliczając do jednej klasy te i tylko te wektory, które przez przesunięcie równoległe przechodzą na siebie. Mówiąc inaczej tworzymy klasę wektorów równoważnych. Każda taka klasa scharakteryzowana jest już nie przez położenie, lecz przez zwrot, kierunek i długość swoich elementów. Takie klasy wektorów równoważnych nazywamy *wektorami swobodnymi*. Wektor swobodny nie ma określonego położenia na płaszczyźnie (często mówi się, że może być położony w dowolnym miejscu płaszczyzny) ma jedynie długość i zwrot.

Zasada abstrakcji jest twierdzeniem, pozwalającym uogólnić postępowanie takie jak opisane w przykładzie 2, na dowolny zbiór w którym określona jest relacja równoważności.

TWIERDZENIE 1. (Zasada abstrakcji). *Jeżeli w zbiorze X określona jest relacja równoważności „$\equiv$", to klasy (podzbiory)*: $[x]$, $[y]$, ... *(zwane klasami abstrakcji relacji „$\equiv$"), określone następująco*:

$$z \in [x] \text{ wtedy i tylko wtedy, gdy } z \equiv x,$$

spełniają następujące warunki:

1. *każda klasa jest niepusta*;
2. *suma wszystkich klas daje zbiór X*;
3. *każde dwie klasy są albo rozłączne albo identyczne*;
4. *dwie klasy są identyczne $[x] = [z]$ wtedy i tylko wtedy, gdy $x \equiv z$.*

Dowód. Niech w zbiorze X określona będzie relacja „$\equiv$", równoważności. Weźmy dowolny element $x \in X$ i utwórzmy podzbiór $[x]$ zbioru X, zwany *klasą elementu x*, złożony ze wszystkich tych elementów $y \in X$, które są równoważne x-owi, tj.

$$y \in [x] \text{ wtedy i tylko wtedy, gdy } y \equiv x.$$

Wobec zwrotności relacji „$\equiv$", mamy $x \equiv y$ dla każdego $x \in X$, a więc dla każdego $x \in X$, mamy $x \in [x]$.

Wynika stąd własność 1 klas abstrakcji, że klasy te nie są puste, oraz własność 2, że suma wszystkich klas daje zbiór X. Rzeczywiście każdy element należy do jakiejś klasy, mianowicie do klasy, którą wyznacza.

Udowodnimy teraz, korzystając z własności relacji równoważności (dotychczas wykorzystaliśmy tylko zwrotność), że spełniona jest własność 3 i 4. Udowodnimy mianowicie, że

5. *jeżeli $x \equiv y$, to $[x] = [y]$,*

zaś

6. *jeżeli $x \not\equiv y$, to $[x] \cap [y] = 0$.*

Przypuśćmy, że $x \equiv y$. Niech teraz $z \in [x]$, wtedy jest $z \equiv x$, a wobec przechodniości relacji „$\equiv$", $z \equiv y$, czyli $z \in [y]$. Dowodzi to, że $[x] \subset [y]$. Podobnie jeżeli $z \in [y]$, to $z \equiv y$. Z założenia $x \equiv y$, wobec symetrii relacji jest $y \equiv x$, a wobec przechodniości jest $z \equiv x$, czyli $z \in [x]$. Dowodzi to, że $[y] \subset [x]$. Wynika stąd równość klas $[x] = [y]$, a więc spełniona jest własność 5. Własność 6 wynika z następujących rozważań. Gdyby $[x] \cap [y]$ było niepuste, to istniałby element z taki, że $z \in [x]$ i $z \in [y]$. Stąd $z \equiv x$ i $z \equiv y$, a wobec symetrii i przechodniości $x \equiv y$. Stąd jeżeli $x \not\equiv y$, to przekrój $[x] \cap [y]$ musi być pusty.

Własności 3 i 4 wynikają natychmiast z własności 5 i 6.

Zbadamy teraz jak wyglądają klasy abstrakcji dla różnych relacji równoważności. Dla relacji równości „$=$", klasy abstrakcji będą jednoelementowe, gdyż wszystkie elementy jakiejś klasy będą musiały być równe. W drugim skrajnym przypadku będzie tylko jedna klasa abstrakcji złożona z całego zbioru X. Będzie to miało miejsce wtedy, gdy relacją równoważności, dla której tworzymy klasy abstrakcji będzie relacja zachodząca między każdymi dwoma elementami zbioru X. (Por. zadania 1 poprzedniego paragrafu).

PRZYKŁAD 3. Rozpatrzmy teraz podział zbioru X na jakieś podzbiory (klasy) $X = K_1 \cup K_2 \cup K_3 \cup \ldots$ spełniające warunki 1, 2 i 3 twierdzenia 1. Określmy w zbiorze X relację $\underset{k}{\equiv}$ następująco:

$$x \underset{k}{\equiv} y \text{ wtedy i tylko wtedy, gdy } x \text{ i } y \text{ należą do tej samej klasy.}$$

Tak określona relacja jest relacją równoważności. Rzeczywiście wobec 2, element x należy do jakiejś klasy. Powiemy to nieco inaczej, choć trochę sztucznie, x i x należą do tej samej klasy, więc $x \underset{k}{\equiv} x$. Prawo zwrotności

musi więc zachodzić. Jeżeli x i y należą do tej samej klasy, to y i x też należą do tej samej klasy, więc zachodzi prawo symetrii:

$$\text{jeżeli } x \underset{k}{\equiv} y, \text{ to } y \underset{k}{\equiv} x.$$

Niech teraz $x \underset{k}{\equiv} y$ i $y \underset{k}{\equiv} z$. Z definicji relacji „$\underset{k}{\equiv}$" wynika, że x i y należą do jakiejś klasy K_i, y i z zaś do jakiejś klasy K_j. Udowodnimy, że $K_i \equiv K_j$. Rzeczywiście ponieważ $z \in K_i \cap K_j$, więc zgodnie z własnością 3 klasy K_i i K_j muszą być identyczne. Wynika stąd, że elementy x i z należą do tej samej klasy, więc $x \underset{k}{\equiv} z$. Dowodzi to przechodniości relacji „$\underset{k}{\equiv}$".

Z przykładu tego wynika, że dowolne podzbiory spełniające warunki 1, 2 i 3, są klasami abstrakcji jakiejś relacji równoważności.

TWIERDZENIE 2. *Jeżeli dla relacji równoważności „$\equiv$" w zbiorze X określimy $K_1 = [x]$, $K_2 = [y]$, ..., to relacja „$\underset{k}{\equiv}$" będzie identyczna z relacją „$\equiv$". Na odwrót, jeżeli dla relacji „$\underset{k}{\equiv}$" wyznaczonej przez podział na klasy taki jak w przykładzie 3, określimy klasy abstrakcji $[x]$, $[y]$, ..., to będą one identyczne z klasami $K_1, K_2, \ldots$*

Zasada abstrakcji ma dla matematyki wielkie znaczenie. Pozwala ona z elementów jakiegoś zbioru przedmiotów, w którym jest określona relacja równoważności, tworzyć nowe obiekty — klasy abstrakcji tej relacji, utożsamiając wszystkie przedmioty równoważne.

Utożsamiając na przykład pośród zbiorów skończonych wszystkie zbiory równoliczne otrzymamy nowe obiekty — liczby elementów zbioru, czyli liczby naturalne. (Por. zad. 2 poprzedniego paragrafu, i zadanie 4 tego paragrafu). Liczby naturalne są przy takim rozumieniu pewnymi abstrakcyjnymi — wspólnymi — własnościami zbiorów skończonych równolicznych. Zbiory takie mają tylko jedną własność wspólną — mają tyle samo elementów.

Podobnie utożsamiając na płaszczyźnie wszystkie figury przystające otrzymamy pewne abstrakcje — figury geometryczne niezależne od położenia na płaszczyźnie. Własności takich figur stanowią przedmiot badania geometrii. Są to własności wspólne wszystkim figurom przystającym — te własności, które nie zmieniają się przy przesunięciu lub obrocie figury, np. pole, lub ilość boków są własnościami geometrycznymi wielokątów, ale położenie wielokąta na płaszczyźnie nie jest jego własnością geometryczną.

Streszczenie

Ustaliliśmy związek między relacjami równoważności w zbiorze X a podziałami zbioru X na klasy $K_1, K_2, \ldots$ takie, że suma $K_1 \cup K_2 \cup \ldots = X$ oraz $K_1, K_2, \ldots$ są to klasy niepuste i rozłączne. Dla elementu $x \in X$ oznaczamy przez $[x]$ zbiór takich $y \in X$, że $y \equiv x$. Zbiór $[x]$ nazywamy *klasą abstrakcji* elementu x, względem relacji „$\equiv$". Różne klasy abstrakcji relacji równoważności są niepuste i rozłączne i dają w sumie cały zbiór X. Na odwrót każdy podział X na takie klasy wyznacza pewną relację równoważności.

Zadania

1. Udowodnić twierdzenie 2.

2. Opierając się na przykładzie 2 i twierdzeniu 2 obliczyć liczbę różnych relacji równoważności w zbiorze n-elementowym.

3. Weźmy n zmiennych $p_1, \ldots, p_n$ przebiegających zbiór zdań. Weźmy wszystkie formuły poprawne $f(p_1, \ldots, p_n)$ zbudowane z tych zmiennych, za pomocą funktorów zdaniowych implikacji i negacji. Oznaczmy ten zbiór wyrażeń przez X. Udowodnić, że relacja „$\approx$" określona dla $f(p_1, \ldots, p_n)$, $g(p_1, \ldots, p_n) \in X$ następująco:

$f(p_1, \ldots, p_n) \approx g(p_1, \ldots, p_n)$ wtedy i tylko wtedy, gdy wyrażenie $f(p_1, \ldots, p_n) \equiv g(p_1, \ldots, p_n)$ jest tautologią, jest relacją równoważności.

Udowodnić, że liczba klas abstrakcji tej relacji w zbiorze X jest równa 2^n. Czym są te klasy abstrakcji?

4. Czym są klasy abstrakcji relacji równoliczności rwl, określonej w zadaniu 2 poprzedniego paragrafu? Ile jest tych klas w przypadku, gdy zbiór A zawiera n elementów?

5. Udowodnić, że dla dwóch relacji równoważności ich iloczyn (por. zad. 3, § 32) jest relacją równoważności. Jak będą wyglądać klasy abstrakcji tego iloczynu?

6. Podać klasy abstrakcji relacji równoważności podanych w zadaniu 3 poprzedniego paragrafu.

7. Ile jest klas abstrakcji relacji określonej w zad. 4 z poprzedniego paragrafu?

§ 35. RELACJE PORZĄDKU

W paragrafie tym rozpatrzmy relacje mające nieco odmienne własności niż relacje równoważności. Rozważmy trzy własności relacji R określonej w jakimś zbiorze X:

1. *Zwrotność*: dla każdego $x \in X$: xRx.
2. *Antysymetria*: dla każdego $x, y \in X$: jeżeli xRy i yRx, to $x = y$.
3. *Przechodniość*: dla każdych $x, y, z \in X$: jeżeli xRy i yRz, to xRz.

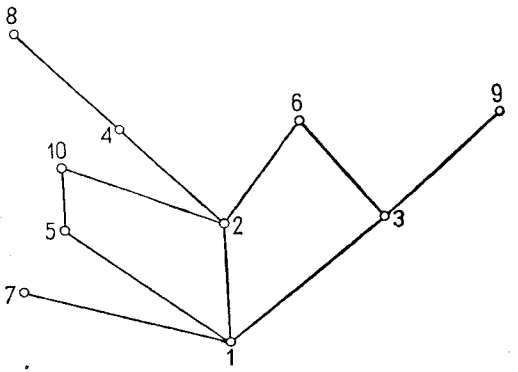
Przyjmijmy następującą definicję:

DEFINICJA 1. Każdą relację R określoną w jakimś zbiorze i mającą własności 1, 2, i 3, (zwrotności, antysymetrii i przechodniości) nazywać będziemy *relacją porządku*. (Często używa się również terminu *porządek częściowy*).

Własności 1 i 3 już znamy, są to własności wspólne relacjom równoważności i relacjom porządku. Własność 2 jest inna niż własność symetryczności. Mówi ona, że jeżeli element x jest w relacji R do y i jednocześnie y w relacji R do x to oba elementy muszą być równe.

Podamy teraz przykłady relacji porządku.

PRZYKŁAD 1. Relacja podzielności $x|y$ w jakimś zbiorze złożonym z liczb naturalnych jest relacją porządku. Prawo zwrotności zachodzi, gdyż każdy element dzieli siebie samego: $x|x$. Prawo przechodniości również, gdyż jeżeli $x|y$ i $y|z$, to $x|z$. Antysymetria wynika ze znanego prawa



Rys. 4. Graf relacji częściowego porządku w zbiorze $X = \{1, 2, \ldots, 10\}$

arytmetyki liczb naturalnych, mówiącego, że jeżeli $x|y$ i $y|x$, to liczby naturalne x i y są równe. Przyjrzyjmy się rysunkowi 4. Na rysunku tym pokazany jest graf relacji porządku, dla relacji podzielności w zbiorze X złożonym z liczb naturalnych od 1 do 10:

$$X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$$

narysowany trochę inaczej niż na rysunku 3. Dla prostoty pominięto w ogóle strzałki z x do x. Kierunku strzałek nie wyróżniano, przyjmując że jest zgodny z wznoszeniem do góry.

Graf ten odczytuje się następująco: x dzieli y wtedy i tylko wtedy gdy x jest równe y lub gdy z x do y można przejść, poruszając się w górę po jakiejś łamanej. Z grafu tego widać, że np. $2|6$, gdyż po łamanej grafu możemy się poruszać od 2 do 6 w górę. Ale na przykład 2 nie dzieli 3 gdyż nie możemy przejść od 2 do 3 idąc stale do góry. Widać stąd, że dla relacji R porządku mogą istnieć pary elementów nieporównywalnych takich, że nie zachodzi ani xRy, ani yRx. W przypadku relacji podzielności jest wiele takich par, np. pary: 2,3; 4,10; 10,4; i inne.

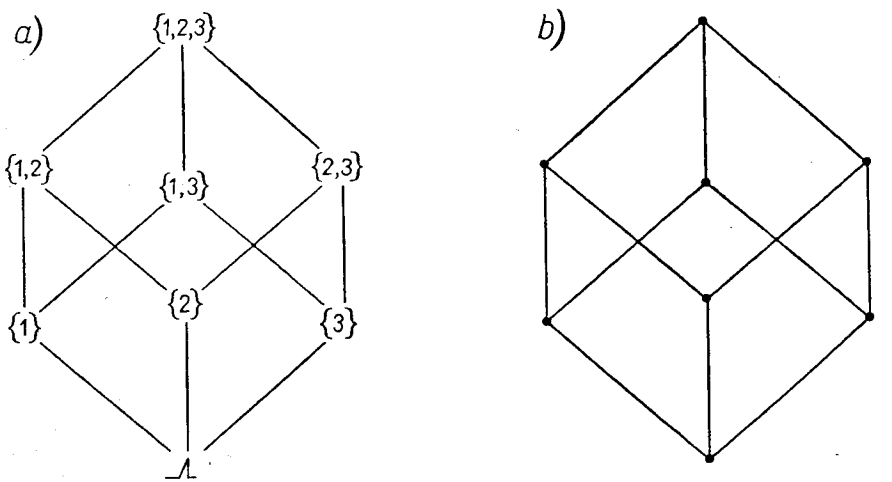
PRZYKŁAD 2. Niech X będzie zbiorem podzbiorów ustalonego zbioru A. Elementy $x, y, z, \ldots$ zbioru X będą więc podzbiorami zbioru A. Relacja inkluzji czyli zawierania tych podzbiorów: $x \subset y$ opisana w rozdziale V będzie relacją częściowego porządku w zbiorze X. Prawa:

prawo zwrotności: $x \subset x$;

prawo antysymetrii: jeżeli $x \subset y$ i $y \subset x$, to $x = y$;

prawo przechodniości: jeżeli $x \subset y$ i $y \subset z$, to $x \subset z$, są bowiem prawami rachunku zbiorów.

Graf tej relacji, w przypadku gdy zbiór $A = \{1, 2, 3\}$ jest zbiorem trójelementowym, podany jest na rysunku 5. Grafu tego nie można przed



Rys. 5. Graf relacji inkluzji zbioru X podzbiorów zbioru $A = \{1, 2, 3\}$

stawić płasko, nie rysując przecięć (rys. 5a) natomiast można przedstawić przestrzennie przez krawędzie sześcianu postawionego na jednym z wierzchołków (rys. 5b). Z grafu widać, że istnieją elementy nieporównywalne.

PRZYKŁAD 3. Rozpatrzmy relację $x \leqslant y$ mniejszości lub równości w jakimś zbiorze liczb. Relacja ta jest relacją porządku, gdyż własności takie jak:

zwrotność: $x \leqslant x$,

antysymetria: jeżeli $x \leqslant y$ i $y \leqslant x$, to $x = y$,

przechodniość: jeżeli $x \leqslant y$ i $y \leqslant z$, to $x \leqslant z$,

wynikają z własności relacji „x mniejsze lub równe y" dla liczb. Relacja ta różni się jednak swoimi własnościami od relacji pokazanych w przykładach 2 i 3. W zbiorze liczb nie ma bowiem elementów nieporównywalnych, gdyż relacja „$\leqslant$" spełnia następujące prawo, zwane *prawem spójności*

dla dowolnych liczb x i y jest: $x \leqslant y$ lub $y \leqslant x$.

Przyjmuje się następującą definicję:

DEFINICJA 2. Relację R porządku w zbiorze X nazywamy *relacją liniowego porządku* (1), jeżeli spełnia następującą własność spójności:

4. dla każdych x i y: xRy lub yRx.

Przykładem relacji liniowego porządku jest relacja „mniejsze lub równe" z przykładu 3. Innym ważnym przykładem relacji liniowego porządku jest tak zwane *uporządkowanie leksykograficzne*, które wyjaśnimy w poniższym przykładzie.

PRZYKŁAD 4. Rozpatrzmy zbiór X ciągów $x = x_1, x_2, x_3, \ldots$, których wyrazami są liczby i określmy relację $x \prec y$ dla dwóch ciągów $x = x_1, x_2, x_3, \ldots, x_n$; $y = y_1, y_2, y_3, \ldots, y_n$ następująco:

$x \prec y$ wtedy i tylko wtedy, gdy albo $x_i = y_i$ dla każdego i (tzn. $x = y$), albo istnieje takie i, że

$$x_1 = y_1, \quad x_2 = y_2, \quad \ldots, \quad x_{i-1} = y_{i-1} \quad \text{i} \quad x_i < y_i.$$

(1) W literaturze używa się również często słów: częściowy porządek na oznaczanie porządku, oraz porządek na oznaczanie porządku liniowego. Nazwa porządek liniowy pochodzi z tego, że wykres relacji tego porządku będzie jedną linią bez odgałęzień.

Na przykład w myśl tej definicji będzie:

$$1, 3, 2, 7, 5, 4 \prec 1, 3, 5, 2, 1, 5.$$

gdyż $x_1 = 1 = y_1$, $x_2 = 3 = y_2$, $x_3 = 2 < y_3 = 5$, ...

Z definicji wynika natychmiast $x \prec x$. Sprawdźmy przechodniość: Relacja $x \prec y$ mówi, że istnieje i dla którego

$$x_1 = y_1, \quad \ldots, \quad x_{i-1} = y_{i-1} \quad \text{i} \quad x_i < y_i.$$

Relacja $y \prec z$ mówi, że istnieje j, dla którego

$$y_1 = z_1, \quad \ldots, \quad y_{j-1} = z_{j-1} \quad \text{i} \quad y_j < z_j.$$

Położmy $k = \min(i, j)$, wtedy będzie

$$x_1 = z_1, \quad x_2 = z_2, \quad \ldots, \quad x_{k-1} = z_{k-1}.$$

Dla wyrazów x_k, y_k i z_k będzie zachodzić jedna z trzech możliwości:

$$x_k = y_k \quad \text{i} \quad y_k < z_k,$$

$$x_k < y_k \quad \text{i} \quad y_k < z_k,$$

$$x_k < y_k \quad \text{i} \quad y_k = z_k,$$

czyli zawsze będzie $x_k < z_k$. To dowodzi, że $x \prec z$, czyli prawo przechodniości relacji „$\prec$" zachodzi.

Łatwo zauważyć, że jeżeli $x \prec y$ i $y \prec x$ to dla każdego i musi być $x_i = y_i$, czyli $x = y$. Dowodzi to własności antysymetrii.

Dla wykazania, że „$\prec$" jest relacją liniowego porządku trzeba pokazać jeszcze spójność relacji „$\prec$". Jeżeli $x = y$, to zarówno $x \prec y$ jak i $y \prec x$. Jeżeli $x \neq y$, to oznaczmy przez i najmniejsze takie j, że $x_j \neq y_j$. Jeżeli teraz $x_i < y_i$, to $x \prec y$, jeżeli zaś $y_i < x_i$, to $y \prec x$.

A więc zachodzi spójność, gdyż dla dowolnych x i y jedna z relacji:

$$x \prec y \quad \text{lub} \quad y \prec x$$

musi zachodzić.

Uporządkowanie to nazywamy *uporządkowaniem leksykograficznym*. Jest to takie uporządkowanie jak uporządkowanie słów w słowniku.

Na zakończenie rozpatrzmy jeszcze jeden przykład.

PRZYKŁAD 5. Niech R będzie jakąś relacją określoną w zbiorze X. Zdefiniujemy w tym zbiorze nową relację R* w następujący sposób: Dla dowolnych $x, y \in X$: $x\mathrm{R}^*y$ wtedy i tylko wtedy, gdy $y\mathrm{R}x$. Relację tę nazywamy *relacją odwrotną* do R. Łatwo widać, że (R*)* daje z powrotem relację R. Na przykład relacja „$\leqslant^*$" jest relacją „$\geqslant$", relacja „$<^*$" jest relacją „$>$", jeśli zaś relacja R jest relacją równoważności, to R* jest tą samą relacją R.

Z postaci aksjomatów relacji porządku łatwo wynika, że jeżeli R jest relacją porządku, to relacja R* jest również relacją porządku. Ponadto jeżeli R jest relacją porządku liniowego, to R* jest również relacją porządku liniowego.

Streszczenie

Podaliśmy pojęcie relacji porządku i relacji liniowego porządku. Omówiliśmy przykłady relacji porządku podzielności liczb naturalnych, inkluzji zbiorów, mniejszości równości dla liczb oraz porządku leksykograficznego. Dwa ostatnie są przykładami relacji porządku liniowego

Zadania

1. Opisać wszystkie relacje porządku na zbiorze trójelementowym. Podać ich grafy. Które spośród nich są relacjami porządku liniowego?

2. Podać graf relacji „$\leqslant$" w zbiorze liczb $x = \{1, 2, 3, \ldots, 10\}$ i porównać z grafem podanym na rysunku 4.

3. Pokazać, że relacja mniejszości w zbiorze liczb nie jest relacją porządku. Por. zad. 5.

4. Udowodnić, że na zbiorze n-elementowym można określić dokładnie $1 \cdot 2 \cdot 3 \cdot \ldots \cdot n = n!$ różnych relacji porządku liniowego.

5. Udowodnić, że jedyną relacją R porządku, taką że $\mathrm{R} = \mathrm{R}^*$ jest równość. Udowodnić, że jeżeli R jest relacją porządku w zbiorze X, to relacja S zdefiniowana następująco:

dla dowolnych $x, y \in X$: $x\mathrm{S}y$ wtedy i tylko wtedy, gdy $x\mathrm{R}y$ lub $x = y$ jest identyczna z R.

6. Dla danej relacji porządku R określonej w zbiorze X, zdefiniujemy relację T następująco:

dla dowolnych $x, y \in X$: $x\mathrm{T}y$ wtedy i tylko wtedy, gdy $x\mathrm{R}y$ i $x \neq y$.

Udowodnić, że relacja T spełnia następujące prawa:

1') *przeciwzwrotności*: dla każdego $x \in X$, nieprawda, że $x\mathrm{T}x$.

2') *przeciwsymetrii*: dla dowolnych $x, y \in X$: jeżeli $x\mathrm{T}y$, to nieprawda, że $y\mathrm{T}x$.

3') *przechodniości*: dla dowolnych $x, y, z \in X$: jeżeli $x\mathrm{T}y$ i $y\mathrm{T}z$, to $x\mathrm{T}z$.

Tak określoną relację nazywamy czasem *relacją ostrego porządku.*

7. Niech R_1 i R_2 będą relacjami porządku określonymi odpowiednio na zbiorach X_1 i X_2 rozłącznych. Niech zbiór X będzie sumą zbiorów X_1 i X_2. Określmy na zbiorze X relację R w następujący sposób:

I. dla $x, y \in X_1$: $x\mathrm{R}y$ wtedy i tylko wtedy, gdy $x\mathrm{R}_1y$,

II. dla $x, y \in X_2$: $x\mathrm{R}y$ wtedy i tylko wtedy, gdy $x\mathrm{R}_2y$,

III. dla $x \in X_1$, $y \in X_2$: $x\mathrm{R}y$ zachodzi zawsze;
dla $x \in X_2$, $y \in X_1$: $x\mathrm{R}y$ nie zachodzi nigdy.

Udowodnić, że relacja R jest relacją porządku w zbiorze X.

8. Określić uporządkowanie leksykograficzne ciągów nieskończonych i udowodnić, że jest to relacja porządku liniowego.

9. Udowodnić, że jeżeli R jest relacją liniowego porządku w zbiorze X, to złożenie (por. zad. 4, § 26, tego rozdziału) $\mathrm{R} \cdot \mathrm{R}^*$ jest relacją pełną zachodzącą między każdymi dwoma elementami zbioru X.

§ 36. SIATKI

Niech X będzie zbiorem uporządkowanym przez jakąś relację porządku, którą w dalszym ciągu oznaczać będziemy przez „$\leqslant$". O relacji tej nie będziemy jednak wcale zakładać, że jest ona relacją porządku liniowego. Nie należy więc jej mylić z relacją mniejsze lub równe dla liczb, wprowadzoną w przykładzie 3 z poprzedniego paragrafu oznaczaną tam tym samym symbolem.

Wprowadźmy następującą definicję.

DEFINICJA 1. Element c nazywać będziemy *ograniczeniem górnym* pary elementów a i b, jeżeli $a \leqslant c$ i $b \leqslant c$. Podobnie jeżeli $d \leqslant a$ i $d \leqslant b$, to element d nazywać będziemy *ograniczeniem dolnym* pary elementów a i b.

Na przykład dla zbioru uporządkowanego podanego w przykładzie 1 z poprzedniego paragrafu (por. rys. 4) ograniczeniem górnym pary $a = 2$ i $b = 3$ będzie element $c = 6$. Ograniczeniem dolnym będzie element $d = 1$. Podobnie dla pary $a = 4$, $b = 8$, ograniczeniem górnym będzie element $c = 8$. Ograniczenia dolne będą trzy. Jedno $d_1 = 4$, drugie $d_2 = 2$, trzecie $d_3 = 1$. Para $a = 5$ i $b = 7$ będzie miała jedno ograniczenie dolne $d = 1$, ale nie będzie miała ograniczenia górnego.

Jeżeli relacja „$\leqslant$" jest relacją porządku liniowego w zbiorze X, to każda para (a, b) ma co najmniej jedno ograniczenie górne i co najmniej jedno ograniczenie dolne. Mianowicie z warunku spójności wynika, że $a \leqslant b$ lub $b \leqslant a$. W przypadku $a \leqslant b$ ograniczeniem dolnym będzie a, górnym zaś b. W przypadku $b \leqslant a$ będzie przeciwnie.

DEFINICJA 2. Ograniczenie górne c pary (a, b) nazywamy *kresem górnym* tej pary, jeżeli dla każdego ograniczenia górnego c_1 tej pary jest $c \leqslant c_1$. Podobnie ograniczenie dolne d pary (a, b) nazywamy *kresem dolnym* tej pary, jeżeli dla każdego ograniczenia dolnego d_1 tej pary jest $d_1 \leqslant d$.

Z definicji wynika, że dla każdej pary (a, b) zbioru uporządkowanego X może istnieć co najwyżej jeden kres górny i co najwyżej jeden kres dolny. Kresy te o ile istnieją oznaczamy przez $a \cup b$ i $a \cap b$. Można łatwo udowodnić, że jeżeli kresy te istnieją, to $a \cap b \leqslant a \cup b$.

TWIERDZENIE 1. *Dla kresów, jeżeli one istnieją, zachodzą następujące prawa*:

1. $x \cap x = x$, $\quad x \cup x = x$,
2. $x \cap y = y \cap x$, $\quad x \cup y = y \cup x$,
3. $x \cap (y \cap z) = (x \cap y) \cap z$, $\quad x \cup (y \cup z) = (x \cup y) \cup z$,
4. $x \cap (x \cup y) = x$, $\quad x \cup (x \cap y) = x$

zwane *prawami idempotentności, przemienności, łączności i pochłaniania.*

Z poprzednich rozważań tego paragrafu widzieliśmy, że kresy nie zawsze muszą istnieć. Przykład 2 z poprzedniego paragrafu był o tyle ciekawy, że dla relacji „$\subset$" inkluzji w zbiorze wszystkich podzbiorów ustalonego zbioru, kres górny $a \cup b$ i kres dolny $a \cap b$ zawsze istnieją. Kresem górnym jest suma zbiorów, kresem zaś dolnym ich przekrój.

DEFINICJA 3. Zbiory uporządkowane, w których dla każdej pary (a, b) istnieją kres górny $a \cup b$ i kres dolny $a \cap b$, nazywają się *siatkami* lub *strukturami*[1].

Strukturami są więc np. zbiory uporządkowane liniowo (por. zad. 3). Strukturą jest również zbiór wszystkich podzbiorów ustalonego zbioru,

[1]) Termin siatka jest wiernym tłumaczeniem słowa angielskiego „lattice" i rozpowszechnia się coraz bardziej. Mimo, że dotąd w powszechnym ujęciu jest termin struktura użyliśmy w książce terminu siatka, gdyż termin ten nie powoduje niejednoznaczności.

częściowo uporządkowany przez relację inkluzji. Inne przykłady struktur są podane w zadaniach 4 i 6.

Podaliśmy tutaj tylko definicję struktury. Teoria struktur stanowi obszerną dziedzinę badań matematycznych, mającą rozliczne i ważne zastosowanie. Teorii tej poświęcona jest monografia Birkhoffa [1948]. Z elementami teorii struktur najłatwiej jest zapoznać się z książki Birkhoffa i Mc. Lane'a lub nieco obszerniej z książki Jacobsona, Rasiowej i Sikorskiego, Hermesa [1955].

Streszczenie

Podaliśmy definicję kresu górnego i kresu dolnego pary elementów zbioru uporządkowanego. Podaliśmy definicję siatki (struktury).

Zadania

1. a) Udowodnić, że jeżeli c jest ograniczeniem górnym pary (a, b), zaś $c \leqslant c_1$, to c_1 jest również ograniczeniem górnym pary (a, b).

b) Podobnie jeżeli d jest dolnym ograniczeniem pary (a, b) zaś $d_1 < d$, to d_1 jest dolnym ograniczeniem pary (a, b).

c) Uzasadnić, że w każdym zbiorze uporządkowanym para (a, a) ma ograniczenie dolne i ograniczenie górne.

2. a) Udowodnić, że kres górny jest co najwyżej jeden.

Wskazówka: Jeżeli c i c_1 są kresami pary (a, b), to musi być $c < c_1$ i $c_1 < c$.

b) Udowodnić, że kres dolny jest co najwyżej jeden.

c) Udowodnić, że jeżeli $a \cap b$ i $a \cup b$ istnieją, to

$$a \cap b < a < a \cup b,$$

$$a \cap b < b < a \cup b.$$

d) Wywnioskować stąd, że jeżeli $a \cap b = a \cup b$, to

$$a = b = a \cap b = a \cup b.$$

3. Udowodnić, że jeżeli relacja „$<$" jest relacją porządku liniowego w zbiorze X, to zawsze istnieje $a \cup b$ i $a \cap b$. Wtedy $a \cup b = \max(a, b)$, zaś $a \cap b = \min(a, b)$. (Przypominamy, że jeżeli „$<$" jest relacją porządku liniowego, to zawsze $a < b$ lub $b < a$).

4. Udowodnić, że jeżeli w zbiorze N^* liczb naturalnych dodatnich przyjąć za relację $x \leqslant y$ relację $x|y$ (x dzieli y), to zawsze istnieją $a \cup b$ i $a \cap b$ oraz

$$a \cup b = \text{n.w.w.}(a, b),$$

$$a \cap b = \text{n.w.d.}(a, b).$$

W nazwie „najmniejsza wspólna wielokrotność" (n.w.w.) wyraz „ najmniejsza" odnosi się do relacji podzielności określonej powyżej, a nie do wielkości liczb. Podobnie w nazwie największy wspólny dzielnik (n.w.d.)

5. Udowodnić twierdzenie 1.

6. Niech $\mathscr{R}(X)$ będzie zbiorem wszystkich relacji zachodzących w zbiorze X. Określmy między dwoma relacjami $R, S \in \mathscr{R}(X)$ relację $R \leqslant S$ w następujący sposób:

$R \leqslant S$ wtedy i tylko wtedy, gdy dla każdych $x, y \in X$: jeżeli xRy, to xSy.

a) Udowodnić, że relacja „$\leqslant$" jest relacją porządku w zbiorze $\mathscr{R}(X)$.

b) Udowodnić, że zbiór $\mathscr{R}(X)$ tworzy siatkę (strukturę) względem relacji „$\leqslant$".

7. Wskazać na rysunku 4 wszystkie podzbiory czteroelementowe będące siatkami.

8. Udowodnić, że jeżeli zbiór X jest siatką L względem relacji „$\leqslant$", to jeżeli określimy nową relację „$\leqslant^*$" następująco:

$$x \leqslant^* y \text{ wtedy i tylko wtedy, gdy } y \leqslant x,$$

to zbiór X będzie siatką L względem relacji „$\leqslant^*$". *Siatkę L* nazywamy *dualną* względem L.

9. Udowodnić, że zbiór X w którym określone są działania $x \cup y$ i $x \cap y$, które spełniają aksjomaty:

L_1. prawo idempotentności: $x \cap x = x$, $x \cup x = x$;

L_2. prawo przemienności: $x \cap y = y \cap x$, $x \cup y = y \cup x$;

L_3. prawo łączności: $x \cap (y \cap z) = (x \cap y) \cap z$, $x \cup (y \cup z) = (x \cup y) \cup z$;

L_4. prawo pochłaniania: $x \cap (x \cup y) = x$, $x \cup (x \cap y) = x$

jest siatką względem relacji określonej następująco: $a \leqslant b$ wtedy i tylko wtedy, gdy $a \cap b = b$.

Udowodnić, że kres górny (a, b) względem relacji „$\leqslant$" jest równy $a \cup b$, dolny zaś równy $a \cap b$.

10. Udowodnić, że z aksjomatu L_4 z poprzedniego zadania wynika aksjomat L_1.

11. Narysować grafy wszystkich siatek

a) czteroelementowych,

b) pięcioelementowych.

c) W otrzymanych w zadaniach a) i b) zbiorach siatek wskazać pary siatek dualnych względem siebie.

12. Udowodnić, że w każdej siatce L skończonej istnieją dwa elementy wyróżnione 0 i 1 takie, że dla każdego x

(*) $$0 \leqslant x \leqslant 1.$$

Uwaga. Ze wzoru (*) nie wynika, że każda siatka jest zbiorem liniowo uporządkowanym.

13. Strukturę L nazywamy *siatką modularną* (mówi się czasem *dedekindowską*), jeżeli zachodzi prawo zwane *prawem modularności*:

L_5. jeżeli $x \leqslant z$, to $x \cup (y \cap z) = (x \cup y) \cap z$.

a) Wyróżnić, które z grafów siatek pięcioelementowych przedstawiają siatki modularne.

b) Udowodnić, że prawo dualne do prawa modularności jest znowu prawem modularności.

c) Udowodnić, że każda siatka, której elementami są zbiory, działaniami zaś brania kresów, suma $a \cup b$ i przekrój $a \cap b$ zbiorów są modularne.

14. *Siatka L* nazywa się *rozdzielną*, jeżeli zachodzi w niej prawo rozdzielności.

L_6. $x \cap (y \cup z) = (x \cap y) \cup (x \cap z)$.

a) Udowodnić, że każda siatka rozdzielna jest modularna (por. zad. 13).

b) Udowodnić, że w każdej siatce rozdzielnej zachodzą prawa:

L'_6. $x \cup (y \cap z) = (x \cup y) \cap (x \cup z)$.

L''_6. $(x \cap y) \cup (y \cap z) \cup (z \cap x) = (x \cup y) \cap (y \cup z) \cap (z \cup x)$.

c) Udowodnić, że jeżeli siatka spełnia którekolwiek z praw L'_6 lub L''_6, to jest rozdzielna.

d) Udowodnić, że każda siatka, której elementami są zbiory, zaś $a \cup b$ i $a \cap b$ są sumą i przekrojem zbiorów, jest rozdzielna.

15. a) Każdy podzbiór S siatki L uporządkowany liniowo względem relacji „$\leqslant$" nazywa się *łańcuchem*. Udowodnić, że każdy łańcuch jest siatką rozdzielną siatki L.

b) Wywnioskować stąd, że każdy zbiór uporządkowany liniowo przez jakąś relację „$\leqslant$" tworzy siatkę rozdzielną względem tej relacji.

c) Udowodnić, że każda siatka L, taka jak każdy jej podzbiór jest siatką względem relacji „$\leqslant$" tej samej co w L, jest zbiorem uporządkowanym liniowo względem relacji „$\leqslant$".

§ 37. FUNKCJE

W paragrafie tym zajmiemy się definicją funkcji oraz podaniem pewnych pojęć i terminologii związanej z funkcjami.

Przez *funkcję f* na zbiorze elementów X, o wartościach należących do zbioru Y rozumiemy relację f o dziedzinie X i przeciwdziedzinie Y, a więc pewien podzbiór U produktu $X \times Y$ taki, że

1° Dla każdego $x \in X$ istnieje $y \in Y$ takie, że $(x, y) \in U$.

2° Jeżeli $(x, y_1) \in U$ i $(x, y_2) \in U$, to $y_1 = y_2$.

Pierwszy element pary $(x, y) \in U$ nazywa się *argumentem*.

Drugi element pary $(x, y) \in U$ nazywany jest *wartością funkcji f* dla argumentu x. Oznaczamy go przez $f(x)$ i zamiast pisać xfy tak jak zwykliśmy to czynić dla relacji piszemy $y = f(x)$.

Z warunku 1° wynika, że dla każdego argumentu $x \in X$ istnieje wartość funkcji $y = f(x)$. Z warunku 2° wynika, że każdemu argumentowi jest przyporządkowana co najwyżej jedna wartość funkcji. Zbiór U nazywa się *wykresem* funkcji.

Podana definicja funkcji na gruncie teorii relacji jest bardzo intuicyjna i ogólniejsza niż rozumienie funkcji określonej wzorem. Dla określenia funkcji nie potrzeba wzoru, wystarcza podanie wszystkich par (argument, wartość funkcji). Dla określenia funkcji $y = 2x$ nie trzeba podawać wzoru, lecz wystarczy wypisać wszystkie pary $(x, 2x)$, czyli (1, 2), (2,4), (3,6), ... W praktyce nie daje to korzyści, jednak znakomicie ułatwia rozważania teoretyczne i pozwala rozumieć jak ogólne jest pojęcie funkcji.

Często zamiast terminu funkcja f określona na zbiorze X o wartościach ze zbioru Y mówimy o *odwzorowaniu* zbioru X w zbiór Y, lub o jednoznacznym *przyporządkowaniu* elementom ze zbioru X elementów ze zbioru Y. Wszystkie trzy terminy: funkcja, odwzorowanie i przyporządkowanie jednoznaczne są synonimami. Termin przyporządkowanie używa się czasem wymiennie z terminem relacja.

Mówimy o *funkcji różnowartościowej* lub o *przyporządkowaniu wzajemnie jednoznacznym*, jeżeli

3° z tego, że $f(x_1) = f(x_2)$ wynika, że $x_1 = x_2$ (tzn. równym wartościom funkcji odpowiadają równe argumenty).

Mówimy o *funkcji odwzorowującej* zbiór X na zbiór Y lub krócej o odwzorowaniu zbioru X na zbiór Y, jeżeli

4° dla każdego $y \in Y$ istnieje takie x, że $y = f(x)$ (tzn. każdy element zbioru Y jest wartością funkcji dla jakiegoś argumentu).

Funkcje spełniające warunki 3° i 4° nazywają się *funkcjami odwracalnymi*. Wtedy relacja f^{-1} o dziedzinie Y i przeciwdziedzinie X określona następująco:

$yf^{-1}x$ wtedy i tylko wtedy, gdy $y = f(x)$ jest funkcją. Dla zapisania tego, że relacja $yf^{-1}x$ zachodzi dla pary (y, x) piszemy $x = f^{-1}(y)$.

Rzeczywiście warunek 4° zapewnia, że dla każdego $y \in Y$ tzn. y będącego argumentem f^{-1}, istnieje wartość $x = f^{-1}(y)$. Dowodzi to spełnienia warunku 1° dla relacji f^{-1}.

Założenia warunku 2° dla funkcji f^{-1} zapisujemy: $x_1 = f^{-1}(y)$ i $x_2 = f^{-1}(y)$. Daje to $y = f(x_1)$ i $y = f(x_2)$. Stąd wobec warunku 3° mamy $x_1 = x_2$. A więc równym argumentom f^{-1} odpowiadają równe wartości.

Dla dowolnej funkcji f określonej na zbiorze X o wartościach ze zbioru Y zbiór tych $y \in Y$, dla których istnieje $x \in X$ takie, że $y = f(x)$, nazywamy *obrazem* zbioru X przez funkcję f. Obraz ten oznaczamy przez $f(X)$.

Na przykład dla funkcji $y = x^2$ określonej na zbiorze liczb rzeczywistych o wartościach w zbiorze liczb rzeczywistych, obrazem funkcji jest zbiór liczb rzeczywistych nieujemnych. Podobnie dla funkcji $y = x^3$ obrazem zbioru liczb rzeczywistych jest cały zbiór liczb rzeczywistych.

Zawsze $f(X) \subset Y$. Mówimy o *odwzorowaniu w* lub o funkcji przekształcającej zbiór X w zbiór Y, jeżeli chcemy wyrazić fakt, że $f(X) \subset Y$ Jeżeli chcemy specjalnie zaznaczyć, że $f(X) = Y$ mówimy o *odwzorowaniu na* zbiór Y, zgodnie z tym co powiedzieliśmy poprzednio.

Weźmy funkcję f określoną na zbiorze X o wartościach ze zbioru Y i podzbiór A zbioru Y. Określmy na zbiorze A nową funkcję f_A następująco:

$$f_A(x) = f(x) \quad \text{dla } x \in A.$$

Funkcja $f_A(x)$ nazywa się *obcięciem funkcji* f do zbioru A.

Analogicznie dla zbioru X^* zawierającego X, każdą funkcję f^* określoną na całym zbiorze X^* o wartościach ze zbioru Y taką, że

$$f^*(x) = f(x) \quad \text{dla każdego } x \in X$$

nazywamy *przedłużeniem funkcji* f na zbiór X^*. Dla przedłużenia mamy

$$f^*_X(x) = f(x).$$

Udowodnimy teraz następujące twierdzenie.

TWIERDZENIE 1. *Niech zbiór X będzie zbiorem n-elementowym, zbiór Y zaś — zbiorem m-elementowym. Liczba funkcji określonych na zbiorze X, których wartości należą do zbioru Y jest równa m^n.*

Dowód. Dowód przeprowadzimy przez indukcję względem n. Jeżeli zbiór X jest jednoelementowy, to funkcje określone na zbiorze X otrzymamy przyporządkowując jedynemu elementowi X którykolwiek z m elementów zbioru Y. Funkcji na zbiorze jednoelementowym o wartościach z Y będzie więc m, zgodnie z naszym wzorem, gdyż $m^1 = m$.

Prowadząc dowód przez indukcję musimy udowodnić, że jeżeli funkcji określonych na zbiorze n-elementowym, których wartości należą do Y jest m^n, to funkcji określonych na zbiorze $(n+1)$-elementowym, których wartości należą do Y jest m^{n+1}.

Niech zbiór $X_1 = a_1, \ldots, a_n$ ma n elementów, $X = a_1, \ldots, a_n, a_{n+1}$ zaś — $n+1$ elementów. Weźmy funkcję f określoną na X_1. Utwórzmy nową funkcję f^* będącą przedłużeniem f na X. Będzie więc

$$f^*(a_1) = f(a_1), \quad f^*(a_2) = f(a_2), \quad \ldots, \quad f^*(a_n) = f(a_n).$$

Jako wartość $f^*(a_{n+1})$ funkcji f^* na a_{n+1} możemy obrać dowolny spośród m elementów zbioru Y.

Każdą funkcję określoną na X_1 możemy więc przedłużyć do funkcji określonej na X, na m różnych sposobów. Ponadto zauważmy, że jeżeli dwie funkcje f i g były na X_1 różne, to i ich przedłużenia na X będą różne.

Wszystkie przedłużenia funkcji określonych na X_1, będą to wszystkie funkcje określone na X. Funkcji określonych na X jest więc m razy więcej niż funkcji określonych na X_1. Tych ostatnich było wobec założenia indukcyjnego m^n. Więc funkcji określonych na zbiorze X $(n+1)$-elementowym będzie $m \cdot m^n = m^{n+1}$.

Na mocy zasady indukcji dowód twierdzenia został zakończony. Zauważmy jeszcze, że dowód prowadziliśmy, zakładając milcząco, że zarówno n jak i m są różne od zera, czyli że zbiory X i Y są niepuste. Zastanowimy się teraz czy twierdzenia tego nie można uogólnić na przypadek, gdy któryś z tych zbiorów jest pusty.

Jeżeli zbiór Y jest pusty, to nie ma funkcji określonych na zbiorze niepustym, których wartości należą do zbioru pustego. Mówiąc inaczej liczba tych funkcji równa się zero. Jest to zgodne z tym, że dla $n \neq 0$, $0^n = 0$. Jeżeli zbiór X jest pusty (Y zaś jakikolwiek, pusty lub nie), to na zbiorze pustym istnieje zawsze tylko jedna funkcja, której wartości należą do Y. (Funkcją tą będzie *relacja pusta* z X do Y). Jest to zgodne z tym, że $m^0 = 1$, dla każdego m.

Widzimy, że udowodnione twierdzenie jest prawdziwe ogólnie, nie tylko w przypadku, gdy X i Y są niepuste.

Streszczenie

Podaliśmy definicję funkcji, jej argumentów i wartości. Słowa przekształcenie i odwzorowanie używane są jako synonimy słowa funkcja. Podkreśliliśmy różnice w terminologii pomiędzy „odwzorowaniem w" a „odwzorowanie na". Zdefiniowaliśmy funkcje odwracalne i określiliśmy dla takich funkcji odwrotność.

Wprowadziliśmy pojęcie obcięcia i przedłużenia funkcji. Udowodniliśmy, że funkcji określonych na zbiorze m-elementowym o wartościach ze zbioru n-elementowego jest m^n.

Zadania

1. Jeżeli funkcja f odwzorowuje zbiór X w Y, funkcja zaś g — zbiór Y w zbiór Z, to relacja gf o dziedzinie X i przeciwdziedzinie Z określona następująco:

$x\,gf\,z$ wtedy i tylko wtedy, gdy istnieje $y \in Y$ takie, że $x\,f\,y$ i $y\,g\,z$,

jest funkcją. Funkcję gf nazywamy złożeniem funkcji f z funkcją g.

a) Udowodnić, że złożenie funkcji jest funkcją i pokazać, że dla każdego $x \in X$ $gf(x) = g(f(x))$ = wartość funkcji g dla argumentu będącego wartością funkcji f dla x.

b) Udowodnić, że $h(gf) = hg(f)$. Co trzeba przy tym założyć o dziedzinach i przeciwdziedzinach funkcji f, g oraz h?

c) Udowodnić, że złożenie funkcji odwracalnych jest funkcją odwracalną.

2. a) Udowodnić, że jeżeli f jest funkcją odwracalną to $f^{-1}f(x) = x$ dla każdego $x \in X$, zaś $ff^{-1}(y) = y$ dla każdego $y \in Y$.

b) Udowodnić, że dwie funkcje f o wartościach z X i argumentach z Y i g o wartościach z Y i argumentach z X takie, że $gf(x)$ jest tożsamością na X, $fg(y)$ zaś jest tożsamością na Y muszą być odwracalne i $f^{-1} = g$, zaś $g^{-1} = f$.

c) Wywnioskować z punktu b), że jeżeli f jest odwracalne, to f^{-1} jest odwracalne i $(f^{-1})^{-1} = f$ dla każdej funkcji odwracalnej f.

d) Udowodnić, że jeżeli f i g są odwracalne to złożenie fg jest odwracalne oraz $(fg)^{-1} = g^{-1}f^{-1}$.

3. Niech f będzie funkcją określoną na zbiorze X, przyjmującą wartości ze zbioru Y. Określmy dla $A \subset X$ podzbiór $f(A) \subset Y$ zwany obrazem zbioru A w następujący sposób:

$y \in f(A)$ wtedy i tylko wtedy, gdy istnieje $x \in A$ takie, że $y = f(x)$.

Określmy dla $B \subset Y$ podzbiór $f^{-1}(B)$ zbioru X zwany *przeciwobrazem* zbioru B w następujący sposób:

$x \in f^{-1}(B)$ wtedy i tylko wtedy, gdy $f(x) \in B$.

Napis $f^{-1}(B)$ należy rozumieć jako całość. Ma on sens dla dowolnej funkcji f niekoniecznie odwracalnej. Znak f^{-1} w tym zapisie nie oznacza funkcji odwrotnej.

a) Udowodnić, że

$$f^{-1}(f(A)) \supset A, \quad f(f^{-1}(B)) = B.$$

b) Udowodnić związki:

$$f(A_1 \cup A_2) = f(A_1) \cup f(A_2),$$
$$f^{-1}(B_1 \cup B_2) = f^{-1}(B_1) \cup f^{-1}(B_2),$$
$$f(A_1 \cap A_2) \subset f(A_1) \cap f(A_2).$$

(Uwaga. Równość nie musi zachodzić, jeżeli funkcja nie jest równowartościowa).

$$f^{-1}(B_1 \cap B_2) = f^{-1}(B_1) \cap f^{-1}(B_2).$$

c) Udowodnić, że dla złożenia funkcji zachodzą następujące związki:

$$gf(A) = g(f(A)),$$
$$(gf)^{-1}(B) = f^{-1}(g^{-1}(B)).$$

4. a) Podać wszystkie funkcje określone na zbiorze $X = \{a_1, a_2, a_3\}$, których wartości należą do zbioru $Y = \{b_1, b_2, b_3\}$.

b) Podobnie jak w a) przyjmując $X = \{a_1, a_2, a_3, a_4, a_5\}$ i $Y = \{b_1\}$.

5. Metodą podaną w dowodzie twierdzenia 1 udowodnić, że liczba funkcji różnowartościowych określonych na zbiorze n-elementowym X, których wartości należą do zbioru m-elementowego Y, jest równa:

$$\begin{cases} 0, & \text{gdy } m < n, \\ m \cdot (m-1) \cdot (m-2) \cdot \ldots \cdot (m-n+1), & \text{gdy } m \geqslant n. \end{cases}$$

b) Wypisać wszystkie funkcje równowartościowe określone na zbiorze trzyelementowym $X = \{a_1, a_2, a_3\}$, o wartościach w zbiorze $Y = \{b_1, b_2, b_3, b_4\}$.

c) Funkcje wzajemnie jednoznaczne określone na zbiorze n-elementowym X_1, których wartości należą też do tego samego zbioru nazywają się permutacjami n elementów. Uzasadnić, że liczba permutacji zbioru n-elementowego jest równa:

$$1 \cdot 2 \cdot 3 \cdot \ldots \cdot n = n!.$$

d) Wypisać wszystkie permutacje zbioru czteroelementowego.

6. Funkcję φ_A określoną na zbiorze X o wartościach ze zbioru $Y = \{0, 1\}$ taką, że

$$\varphi_A(x) = \begin{cases} 0, & \text{gdy } x \notin A, \\ 1, & \text{gdy } x \in A \end{cases}$$

nazywamy *funkcją charakterystyczną* podzbioru A zbioru X.

a) Udowodnić, że odpowiedniość

$$A \leftrightarrow \varphi_A$$

przyporządkowująca każdemu podzbiorowi A zbioru X, jego funkcję charakterystyczną określoną na X jest wzajemnie jednoznaczna.

b) Wyprowadzić na tej drodze twierdzenie 1 z ostatniego paragrafu poprzedniego rozdziału z twierdzenia 1 tego rozdziału.

Rozdział 7

ALGEBRY BOOLE'A

Algebry Boole'a są ciekawym przykładem kapryśnej i powikłanej linii rozwoju nauki. Nazwa tych algebr pochodzi od nazwiska matematyka i filozofa angielskiego, który około 100 lat temu sformułował podstawowe koncepcje i prawa tychże algebr w związku z badaniami nad formalizacją praw rządzących poprawnymi rozumowaniami. Prawa Boole'a stanowiły ukoronowanie blisko 2000 letnich spekulacji filozofów na temat struktury wnioskowania. W dalszych latach badania różnych własności algebr Boole'a rozwijały się z różną intensywnością, nie wykraczając wszakże w zasadzie poza ramy nakreślone przez ich twórcę. Od około 30 lat datuje się nowe, intensywne zainteresowanie algebrami Boole'a, do czego w znacznym stopniu przyczyniły się niespodziewane ich zastosowania nie tylko w matematyce, lecz również w technice, początkowo w teorii sieci elektrycznych, a później w maszynach matematycznych. W związku z tymi zastosowaniami powstało szereg nowych problemów algebraicznych, ważnych zarówno z matematycznego jak i technicznego punktu widzenia. Na marginesie warto dodać, że wiele z nich do tej pory nie zostało jeszcze zadowalająco rozwiązanych. Tak więc znajomość elementów algebry Boole'a obowiązuje dzisiaj konstruktorów maszyn matematycznych w równym stopniu jak znajomość innych działów matematyki.

Celem naszym nie jest jednakże podanie w tym rozdziale zastosowań algebr Boole'a do rozwiązywania zagadnień technicznych, temat ten jest bowiem szczegółowo przedstawiony w wielu podręcznikach specjalistycznych (patrz np. Mostowskiego, Whitesitta). Chcielibyśmy natomiast, nawiązując do zagadnień poruszanych w rozdziale I i IV, przedstawić zasadnicze idee i koncepcje leżące u podstaw algebr tego rodzaju.

§ 38. DEFINICJA ALGEBR BOOLE'A

Algebrę charakteryzujemy przez podanie zbioru przedmiotów rozważanych w algebrze, elementów wyróżnionych w tym zbiorze, operacji określonych w zbiorze rozważanych przedmiotów oraz pewnych relacji zachodzących między nimi, a także aksjomatów charakteryzujących operacje oraz relacje.

DEFINICJA 1. *Algebrą Boole'a* nazwiemy zbiór X co najmniej dwóch elementów, jeżeli spełnione są następujące warunki:

1. W zbiorze X istnieją dwa elementy wyróżnione, które oznaczymy przez 0 i 1. Zakładamy, że są one różne.

2. W zbiorze X określone są trzy operacje (działania) $A \cup B$, $A \cap B$, A', zwane odpowiednio *sumą boole'owską*, *iloczynem boole'owskim* i *dopełnieniem* — tzn. dla każdych elementów A i B należących do X, również A', $A \cup B$, $A \cap B$ należą do X. Mówimy wtedy, że *zbiór* X jest *zamknięty* ze względu na operacje sumy, iloczynu i dopełnienia.

3. Na elementach zbioru X określona jest relacja równoważności oznaczona przez „=", spełniająca następujący warunek: dla każdych elementów A, B, C należących do zbioru X, jeżeli $A = B$, to również $A' = B'$, $A \cup C = B \cup C$ oraz $A \cap C = B \cap C$.

4. Operacje wymienione w punkcie 2 spełniają następujące aksjomaty(1):

A_1. $A \cup B = B \cup A$,

A_2. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$,

A_3. $A \cup 0 = A$,

A_4. $A \cup A' = 1$,

B_1. $A \cap B = B \cap A$,

B_2. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$,

B_3. $A \cap 1 = A$,

B_4. $A \cap A' = 0$.

Jeżeli zbiór X zawiera tylko dwa elementy, mówimy, że *algebra Boole'a* jest *dwuelementowa*.

(1) Ten układ aksjomatów został zaczerpnięty z Goodsteina [3] Możliwy jest również inny układ aksjomatów algebry Boole'a patrz np. Mostowski A. W., str. 21.

Zwróćmy od razu uwagę na podobieństwa i różnice aksjomatów algebry Boole'a z prawami arytmetyki znanymi nam ze szkoły średniej. Gdybyśmy sumę i iloczyn boole'owski traktowali odpowiednio jako sumowanie i mnożenie liczb naturalnych a 0 i 1 algebry Boole'a jako zero i jeden, to również prawdziwe byłyby aksjomaty A_1, A_2, A_3, B_1, B_3, natomiast pozostałe aksjomaty nie będą zachodziły. Szczególnie interesujące są tu aksjomaty A_2 oraz B_2. Pierwszy z nich jest nazywany prawem rozdzielności dodawania względem mnożenia, drugi zaś prawem rozdzielności mnożenia względem dodawania. O ile pierwsze z tych praw obowiązuje zarówno dla arytmetyki liczb naturalnych jak i dla algebry Boole'a, to drugie natomiast jest ważne tylko dla algebry Boole'a.

Zwróćmy jeszcze uwagę na jeszcze jedną cechę charakterystyczną aksjomatów algebry Boole'a. Zauważmy, że jeżeli w dowolnym aksjomacie grupy A, wszystkie symbole „$\cap$", „$\cup$", „0", „1" zastąpimy według poniższej tabelki

$\cap$	$\cup$
$\cup$	$\cap$
0	1
1	0

to otrzymamy aksjomat grupy B i odwrotnie. Na przykład zastępując w aksjomacie A_2 symbole operacji zgodnie z podaną tabliczką, otrzymamy aksjomat B_2. Własność ta nosi nazwę *zasady dualności* albo *dwoistości algebry*. Zasada dwoistości odnosi się w algebrze Boole'a nie tylko do aksjomatów, ale i do dowolnych równości między termami (zob. r. IV, § 22, str. 84) tej algebry. To znaczy jeżeli w jakiejś równości dwóch wyrażeń algebry Boole'a symbole operacji i stałe zastąpimy według podanej powyżej zasady, to otrzymamy również równość algebry Boole'a. Równości prawdziwe przejdą na równości prawdziwe, spełnione na spełnione, fałszywe zaś na fałszywe.

Streszczenie

Podaliśmy definicję algebry Boole'a wypisując jakie działania muszą być określone i jakie aksjomaty mają spełniać.

Zadania

1. Udowodnić, że zbiór złożony z elementów 0 i 1 z działaniami „$\cup$", „$\cap$" i „$'$" określonymi następująco: $1 \cup 1 = 0 \cup 1 = 1 \cup 0 = 1$, $0 \cup 0 = 0$, $1 \cap 1 = 1$, $1 \cap 0 = 0 \cap 1 = 0 \cap 0 = 0$, $0' = 1$, $1' = 0$, tworzy algebrę Boole'a.

2. Udowodnić, że algebra Boole'a nie może mieć trzech elementów.

Wskazówka: Przypuszczając, że elementami algebry są 0, 1, A, należy rozważyć trzy możliwości: $A' = 0$, $A' = 1$, $A' = A$ (por. zas. 1, § 39 tego rozdziału).

3. Wskazać algebrę Boole'a czteroelementową. Czy algebra Boole'a może być pięcioelementowa?

4. Udowodnić, że jeżeli zbiór B z działaniami „$\cap$", „$\cup$" oraz „$'$", jest algebrą Boole'a, to określając na zbiorze elementów tej algebry nowe działania $\cup^*$, $\cap^*$:

$$x \cup^* y = x \cap y, \quad x \cap^* y = x \cup y$$

dla dowolnych $x, y \in B$ tzn. dla zbioru elementów tej algebry, negację zaś tak jak poprzednio, otrzymamy nową algebrę Boole'a B^*. Pokazać, że $0^* = 1$, $1^* = 0$. Algebra ta nazywa się *algebrą dualną* względem B.

5. Udowodnić, że algebra dualna względem algebry dualnej jest algebrą wyjściową, tj.

$$(B^*)^* = B.$$

§ 39. PRZYKŁADY ALGEBR BOOLE'A

W podanej w poprzednim paragrafie definicji algebr Boole'a nie precyzowaliśmy bliżej czym są elementy algebr Boole'a (tj. elementy zbioru X). Podana definicja ma charakter ogólny, w tym sensie, że elementami algebry Boole'a mogą być jakiekolwiek przedmioty, operacje zaś „$\cup$", „$\cap$" oraz „$'$" mogą być dowolne byleby spełniały aksjomaty algebry. W ten sposób z jednego zbioru przedmiotów przy różnej definicji działań możemy otrzymywać różne algebry Boole'a. Każda taka algebra jest oczywiście pojęciowo czymś różnym, jednakże wszystkie one mają wspólne cechy wyrażone w aksjomatach. Każda algebra Boole'a jest modelem teorii, której aksjomatami specyficznymi są aksjomaty wymienione w punkcie 1-4. Zamiast dowodzić twierdzeń o każdej algebrze Boole'a oddzielnie, dowodzimy twierdzeń korzystając z aksjomatów. Będą one prawdziwe w każdej algebrze Boole'a.

Podamy teraz kilka przykładów algebr Boole'a by zilustrować różnorodność możliwości interpretacji teorii algebr Boole'a.

PRZYKŁAD 1. Tradycyjnie jako przykład algebry Boole'a jest podawana algebra zbiorów. Pojęcie zbioru, działania na zbiorach i równość zbiorów były wyjaśnione w rozdziale V. Przyjmijmy, że elementami zbioru X elementów algebry są podzbiory jakiegoś ustalonego zbioru Z. Jeżeli np. Z jest zbiorem trzech liczb $\{1,2,3\}$, to elementami zbioru X będą wszystkie następujące podzbiory zbioru Z:

$$\begin{aligned} \{1,2,3\} &= 1, & \{1\} &= e_4, \\ \{1,2\} &= e_1, & \{2\} &= e_5, \\ \{1,3\} &= e_2, & \{3\} &= e_6, \\ \{2,3\} &= e_3, & \{0\} &= 0. \end{aligned}$$

Czyli $X = \{\{1,2,3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1\}, \{2\}, \{3\}, \{0\}\} = \{1, e_1, e_2, e_3, e_4, e_5, e_6, 0\}$.

Zbiór Z uważamy również jako podzbiór zbioru Z. Elementami wyróżnionymi zbioru X będą: zbiór pełny $Z = \{1,2,3\}$, który jest jednością w naszej algebrze oraz zbiór pusty $\{0\}$, który jest zerem algebry zbiorów. (Nie mylić jedności zbioru X z elementem 1 zbioru Z). Łatwo sprawdzić, że działania sumy zbiorów, iloczynu zbiorów oraz uzupełnienia zbiorów spełniają aksjomaty algebry Boole'a. Dla ilustracji sprawdzimy niektóre aksjomaty algebry Boole'a:

$$\begin{aligned} \{2,3\} \cap \{1,2,3\} &= \{2,3\}, & &B_3 \\ \{1,3\} \cup \{2\} &= \{1,2,3\}, & &A_4 \\ \{3\} \cap \{1,2\} &= \{0\}, & &B_4 \end{aligned}$$

Oczywiście zbiór Z może zawierać jako elementy jakiekolwiek inne przedmioty niekoniecznie liczby, jak to przyjęliśmy w naszym przykładzie. Zbiór Z może być również zbiorem nieskończonym. W każdym z tych przypadków działania na zbiorach będą spełniały aksjomaty algebry Boole'a.

PRZYKŁAD 2. W rachunku zdań można również skonstruować przykład algebry Boole'a.

Niech X będzie zbiorem wszystkich formuł rachunku zdań zawierających 0, 1 i zmienne $p_1, p_2, \ldots$, zamkniętych ze względu na operacje „$\vee$", „$\&$" oraz „$\sim$". (Oczywiście X jest wtedy zbiorem nieskończonym). Tak więc zbiór X ma postać:

$$X = \{0,1, p_1, p_2, p_3, \ldots, \sim p_1, \ldots, p_1 \vee p_2, \ldots\}.$$

Formuły równoważne będziemy uważać za równe elementy algebry Boole'a. Tak więc elementami algebry Boole'a będą właściwie nie pojedyncze formuły, a klasy formuł równoważnych.

Operacje algebry Boole'a będziemy odpowiednio interpretowali jako spójniki logiczne według tabeli poniżej

$\cup$	$\vee$	(lub)
$\cap$	&	(i)
$'$	$\sim$	(nie)

W ten sposób operacje algebry pozwalają nam tworzyć z jednych formuł nowe formuły, spełniając tym samym warunek, aby argumenty i wyniki tych operacji należały do zbioru X.

0 i 1 są oczywiście interpretowane jako symbol zdania prawdziwego i jako symbol zdania fałszywego.

Pozostaje nam sprawdzić czy przy podanej interpretacji są spełnione aksjomaty algebry Boole'a. Podamy tylko kilka przykładów sprawdzeń aksjomatów, resztę pozostawiając Czytelnikowi jako ćwiczenie. Sprawdzania będziemy dokonywali metodą zero-jedynkową. Na przykład aksjomat A_3 przy podanej interpretacji przechodzi w formułę

$$(p \vee 0) \equiv p. \tag{1}$$

Jeżeli za p podstawimy w formule (1) wartość 0, to otrzymamy w wyniku formułę prawdziwą, podobnie jeżeli podstawimy za p wartość 1. Formuła (1) jest więc zawsze prawdziwa, co znaczy, że aksjomat A_3 przy podanej interpretacji jest spełniony. Podobnie aksjomat B_3 przejdzie na formułę

$$(p \,\&\, 1) \equiv p. \tag{2}$$

Formuła (2) dla p równego 0 jest prawdziwa i podobnie dla p równego 1. Jest więc ona zawsze prawdziwa.

W ten sposób można sprawdzić, że wszystkie aksjomaty algebry Boole'a przejdą w podanej interpretacji w prawa logiki (tautologie logiczne tj. zdania, które są zawsze prawdziwe, niezależnie od wartości logicznej występujących w nich zmiennych zdaniowych). Prawa te będą miały postać:

A_1. $(p \vee q) \equiv (q \vee p)$,

A_2. $(p \vee (q \,\&\, r)) \equiv ((p \vee q) \,\&\, (p \vee r))$,

A_3. $(p \vee 0) \equiv p$,

A_4. $(p \vee \sim p) \equiv 1$,

B_1. $(p \,\&\, q) \equiv (q \,\&\, p)$,

B_2. $(p \,\&\, (q \vee r)) \equiv ((p \,\&\, q) \vee (p \,\&\, r))$,

B_3. $(p \,\&\, 1) \equiv p$,

B_4. $(p \,\&\, \sim p) \equiv 0$.

PRZYKŁAD 3. Niech X będzie zbiorem liczb 1, 2, 3, 5, 6, 10, 15, 30, tj. $X = \{1, 2, 3, 5, 6, 10, 15, 30\}$.[1]

Jako zero zbioru X przyjmiemy liczbę 1, a jako jedność liczbę 30. Przyjmijmy następującą interpretację działań boole'owskich:

$A \cup B$ — najmniejsza wspólna wielokrotność liczb A i B;

$A \cap B$ — największy wspólny dzielnik liczb A i B;

A' — $30/A$ (30 podzielone przez A).

Według przyjętej interpretacji na liczbach należących do zbioru X możemy wykonywać działania, jak to pokazano przykładowo niżej:

$$3 \cup 5 = 15, \quad 2 \cap 3 = 1, \quad 3' = 30/3 = 10,$$
$$2 \cup 6 = 6, \quad 5 \cap 10 = 5, \quad 5' = 30/5 = 6.$$

Prawdziwość aksjomatów A_3 i B_3 przy podanej interpretacji jest oczywista. Aksjomat A_3 mówi bowiem, że dla dowolnej liczby A należącej do X najmniejszą wspólną wielokrotnością tej liczby oraz liczby 1 (zera zbioru X) jest A. Natomiast aksjomat B_3 stwierdza, że dla dowolnej liczby A należącej do X największym wspólnym dzielnikiem A oraz 30 (jedności zbioru X) jest liczba A.

Wykazanie prawdziwości pozostałych aksjomatów jest mniej oczywiste jednakże nie sprawia większych trudności. Pozostawiamy to Czytelnikowi. Należy tylko pamiętać, że najmniejsza wspólna wielokrotność liczb A i B jest iloczynem czynników pierwszych obu liczb A i B (każdy czynnik jest brany tylko raz), największy zaś wspólny dzielnik liczb A i B jest iloczynem czynników pierwszych występujących w obu liczbach A i B. Zwróćmy

[1] Przykład ten jest zaczerpnięty z książki Goodsteina.

jeszcze uwagę, że jeżeli liczba A ma czynniki pierwsze $P(a)$, to czynnikami pierwszymi liczby A' będą wszystkie czynniki pierwsze liczby 30 zbioru X (elementu 1 algebry) — nie występujące w liczbie A. Teraz już sprawdzenie prawdziwości aksjomatów algebry Boole'a nie powinno sprawić trudności. Przejdą one w prawdziwe zdania arytmetyki.

PRZYKŁAD 4. Ciekawym przykładem algebry Boole'a jest *algebra par* (patrz Goodstein i Pawlak). Elementami algebry par będą pary elementów algebry Boole'a. W szczególności zerem algebry par będzie para [0, 1] jednością natomiast — para [1, 0]. Ogólnie elementy algebry par będziemy oznaczali przez $[A, B]$, gdzie A i B są elementami algebry Boole'a. Wprowadzimy następujące definicje operacji na parach:

$$[A, B]' = [A', B'].$$

To znaczy aby znaleźć dopełnienie pary, dopełniamy oba jej elementy. Suma par $[A_1, B_1]$ oraz $[A_2, B_2]$ jest określona w poniższy sposób

$$[A_1, B_1] \cup [A_2, B_2] = [A_1 \cup A_2, B_1 \cap B_2],$$

iloczyn par zaś

$$[A_1, B_1] \cap [A_2, B_2] = [A_1 \cap A_2, B_1 \cup B_2].$$

Powiemy, że pary $[A_1, B_1]$ i $[A_2, B_2]$ są równe: $[A_1, B_1] = [A_2, B_2]$, jeżeli $A_1 = A_2$ i $B_1 = B_2$.

Dla przykładu wykażemy tylko prawdziwość aksjomatów A_3, A_4, B_3, i B_4, pokazanie zaś prawdziwości pozostałych aksjomatów pozostawimy Czytelnikowi.

Aksjomat A_3 w algebrze par będzie miał postać

$$[A, B] \cup [0, 1] = [A, B].$$

Ponieważ

$$[A, B] \cup [0, 1] = [A \cup 0, B \cap 1] = [A, B],$$

więc aksjomat A_3 jest spełniony. Podobnie dla aksjomatu B_3 mamy

$$[A, B] \cap [1, 0] = [A \cap 1, B \cup 0] = [A, B],$$

a więc jest on również prawdziwy. Dla aksjomatów A_4 i B_4 otrzymamy

$$[A, B] \cup [A, B]' = [A, B] \cup [A', B'] = [A \cup A', B \cap B'] = [1, 0],$$

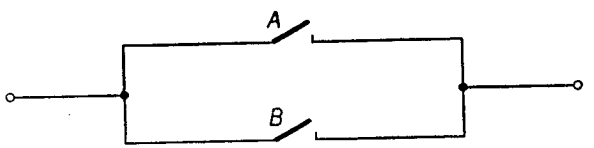
$$[A, B] \cap [A, B]' = [A, B] \cap [A', B'] = [A \cap A', B \cup B'] = [0, 1].$$

Aksjomaty te są więc również prawdziwe. Podobnie możemy sprawdzi prawdziwość pozostałych aksjomatów.

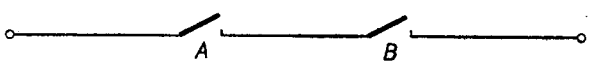
PRZYKŁAD 5. Algebra sieci kontaktowych. Obecny przykład od biega nieco od przykładów rozważanych do tej pory, elementami rozwa żanej algebry będą bowiem nie pojęcia abstrakcyjne jak zbiory, liczb czy funkcje, lecz pewne przedmioty materialne, a mianowicie sieci kontak towe. Niech więc X będzie zbiorem wszystkich możliwych dwubiegu nowych sieci kontaktów, z których każdy może być zamknięty, bąd otwarty. Elementami wyróżnionymi zbioru X będą: sieć składająca si z jednego, ciągle otwartego, kontaktu, którą przyjmiemy za element oraz sieć składająca się z jednego, ciągle zamkniętego kontaktu, któ przyjmiemy za element 1 algebry.

Na sieciach kontaktowych są określone trzy operacje, sumowanie siec mnożenie sieci oraz dopełnienie(1) sieci.

Jeżeli A i B są sieciami należącymi do X, to ich sumą $A \cup B$ jest sie otrzymana przez równoległe połączenie sieci A i B. Na przykład jeże A i B są pojedynczymi kontaktami, to ich suma ma postać



Iloczyn sieci należących do X otrzymamy łącząc sieci A i B szeregow jak to pokazano niżej

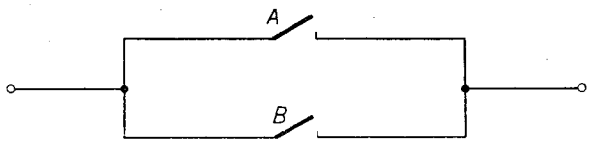


Dopełnienie sieci należącej do X jest operacją nieco bardziej skomp kowaną. Aby dopełnić sieć A, należy zamienić w niej wszystkie zamknię

(1) W języku teorii sieci nie używamy na ogół terminu dopełnienia sieci wzięte z algebry Boole'a — częściej przyjęty jest termin negacja sieci.

kontakty na otwarte i odwrotnie wszystkie otwarte na zamknięte oraz wszystkie połączenia równoległe zamienić na połączenia szeregowe, wszystkie zaś połączenia szeregowe — na równoległe(1).

Dopełnienie sieci

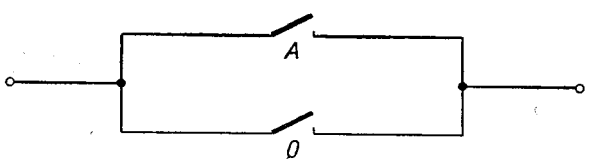


będzie miało postać

Kontakty A i B są kontaktami *biernymi*, natomiast kontakty A' i B' są — *czynnymi*.

Musimy jeszcze określić relację równoważności między sieciami. Dwie *sieci* A i B uważamy za *równoważne* wtedy i tylko wtedy, gdy dla każdych możliwych położeń kontaktów obie sieci przewodzą, bądź nie przewodzą prądu. Łatwo sprawdzić, że określone w ten sposób sieci kontaktów spełniają aksjomaty algebry Boole'a.

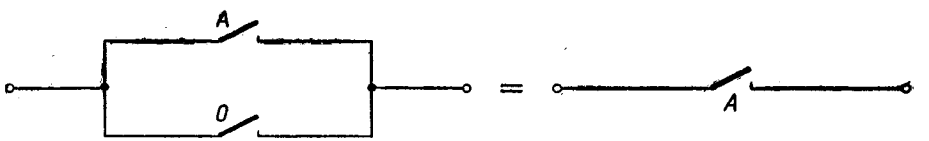
Sprawdźmy dla przykładu aksjomat A_3. Sieć $A \cup 0$, składa się z dwóch kontaktów A oraz 0 połączonych równolegle jak to pokazano niżej



Kontakt 0 jest stale otwarty (nie możemy nim manipulować), natomiast kontakt A (bierny) może być w stanie zamkniętym bądź otwartym. Działanie sieci zależy więc jedynie od kontaktu A i przewodzenie jej, bądź nie przewodzenie następuje zawsze, gdy przewodzi bądź nie przewodzi kontakt A. Sieci $A \cup 0$ oraz A są więc równoważne, czyli

$$A \cup 0 = A$$

lub inaczej



co znaczy, że aksjomat A_3 jest spełniony.

Podana w tym przykładzie interpretacja algebry Boole'a ma duże znaczenie w technice. Pozwala ona projektować sieci kontaktowe o zadanych własnościach, co ma zastosowanie przy projektowaniu różnego rodzaju aparatury łączności, jak np. central telefonicznych, układów automatyki itp. Na temat zastosowania algebry Boole'a do sieci kontaktowych istnieje olbrzymia literatura, w tym wiele podręczników, z których kilka podajemy w spisie literatury: Hohn, Mostowski, Roginskij, Whitesitt.

(1) Ściślej biorąc nie chodzi tu o zmianę położenia kontaktów z zamkniętych na otwarte i odwrotnie, lecz o zmianę kontaktu, który jest w stanie spoczynku otwarty, a w stanie pracy zamknięty (jak np. przycisk dzwonka) — kontaktem, który jest w stanie spoczynku zamknięty, a w stanie pracy otwarty — i na odwrót. Pierwszy rodzaj kontaktów jest czasami nazywany biernym, drugi zaś czynnym. Należałoby więc w definicji sieci dodać, że jest ona zbudowana z dwu rodzajów kontaktów biernych i czynnych.

Streszczenie

Podaliśmy przykłady algebr Boole'a: algebrę zbiorów, algebrę zdań, algebrę par i algebrę sieci kontaktowych.

Zadania

1. Niech X będzie zbiorem wszystkich zbiorów punktów na prostej. Zbiór ten tworzy algebrę Boole'a, której elementami są podzbiory prostej a działaniami „$\cup$", „$\cap$" oraz „$'$", suma, przekrój i uzupełnienie zbiorów.

a) Weźmy podzbiór Y zbioru X (tzn. zbiór podzbiorów prostej) niepusty i taki, że jeżeli $A, B \in Y$, to $A \cup B \in Y$ i jeżeli $A \in Y$, to $A' \in Y$. Udowodnić, że zbiór Y tworzy algebrę Boole'a względem działań określonych tak jak w X. Algebra ta nazywa się *podalgebrą* algebry Boole'a.

b) Udowodnić, że podzbiór niepusty Y tworzy podalgebrę algebry Boole'a, jeżeli dla każdego $A, B \in Y$ również $A \cap B \in Y$ oraz dla $A \in Y$ również $A' \in Y$.

2. a) Sprawdzić, że opisana w przykładzie 4 algebra par spełnia wszystkie aksjomaty algebry Boole'a.

b) Ile elementów będzie miała algebra par, gdy elementy par są brane z algebry Boole'a mającej 2^n elementów?(1)

c) Udowodnić, że dla algebry par $[A, B]$, gdzie elementy A, B są elementami jakiejś algebry Boole'a X, odwzorowane

$$[A, B] \to A$$

1) odwzorowuje algebrę par na całą algebrę X,
2) przeprowadza sumę par na sumę elementów z X,
3) iloczyn par na iloczyn elementów z X,
4) negację pary na negację elementów.

Mówimy, że odwzorowanie to jest *homomorfizmem* algebry par na algebrę X.

d) Udowodnić, że algebra par $[A, B]$, gdzie A, B są elementami jakiejś algebry Boole'a, zawiera podalgebrę izomorficzną z X.

Wskazówka. Algebra składać się będzie z par postaci $[A, A']$

3. Rozpatrzmy zbiór ciągów B^k, (tzn. produkt kartezjański $B \times \ldots \times B$, por. zad. w rozdz. 6)

$$\{A_1, A_2, \ldots, A_k\},$$

gdzie $A_1, A_2, \ldots, A_k$ są elementami jakiejś algebry B. Określmy operacje „$\cup$", „$\cap$" i „$'$" w B^k następująco:

$$\{A_1, A_2, \ldots, A_k\} \cup \{B_1, B_2, \ldots, B_k\} = \{A_1 \cup B_1, A_2 \cup B_2, \ldots, A_k \cup B_k\},$$

$$\{A_1, A_2, \ldots, A_k\} \cap \{B_1, B_2, \ldots, B_k\} = \{A_1 \cap B_1, A_2 \cap B_2, \ldots, A_k \cap B_k\},$$

$$\{A_1, A_2, \ldots, A_k\}' = \{A_1', A_2', \ldots, A_k'\}.$$

a) Udowodnić, że przy takim określeniu działań B^k jest algebrą Boole'a. Jakie ciągi będą elementami wyróżnionymi 0 i 1?

b) Udowodnić, że jeżeli algebra B była skończona i miała 2^n elementów, to algebra B^k ma 2^{nk} elementów.

4. Niech B będzie algebrą Boole'a z zadania 1. Rozpatrzmy zbiór B_n funkcji n zmiennych $X_1, \ldots, X_n$, które są postaci

$$f(X_1, \ldots, X_n) = \bigcup_{e_1, \ldots, e_n} (a_{e_1, \ldots, e_n} \cap X_1^{e_1} \cap \ldots \cap X_n^{e_n}), \quad (2)$$

(1) Jak wynika z rozważań par. § 40 tego rozdziału ilość elementów skończonej algebry Boole'a wyraża się zawsze liczbą będącą potęgą 2.

(2) Często zamiast $A_1 \cup A_2 \cup \ldots \cup A_n$ piszemy $\bigcup_{i \leqslant n} A_i$.

gdzie sumowanie jest po wszystkich układach $e_1, \ldots, e_n$, gdzie $e_i = \begin{cases} 0 \\ 1 \end{cases}$, elementy $a_{e_1, \ldots, e_n}$ są stałymi 0 lub 1 algebry B, X^0 zaś oznacza X', natomiast X^1 oznacza X.

a) Udowodnić, że zbiór funkcji B_n jest zamknięty ze względu na operacje:

$$f(X_1, \ldots, X_n) \cup g(X_1, \ldots, X_n),$$
$$f(X_1, \ldots, X_n) \cap g(X_1, \ldots, X_n),$$
$$f(X_1, \ldots, X_n)'.$$

b) Udowodnić, że dwie funkcje

$$f(X_1, \ldots, X_n) = \bigcup_{e_1, \ldots, e_n} (a_{e_1, \ldots, e_n} \cap X_1^{e_1} \cap \ldots \cap X_n^{e_n})$$

oraz

$$g(X_1, \ldots, X_n) = \bigcup_{e_n, \ldots, e_n} (b_{e_1, \ldots, e_n} \cap X_1^{e_1} \cap \ldots \cap X_n^{e_n})$$

są sobie równe: $f(X_1, \ldots, X_n) = g(X_1 \ldots, X_n,)$, tzn.

$$f(A_1, \ldots, A_n) = g(A_1, \ldots, A_n)$$

dla każdych $A_1, \ldots, A_n \in B$ wtedy i tylko wtedy, gdy

$$a_{e_1, \ldots, e_n} = b_{e_1, \ldots, e_n},$$

dla każdego układu $e_1, \ldots, e_n$ zer lub jedynek.

c) Udowodnić, że zbiór B_n tworzy algebrę Boole'a. Wskazać 0 i 1 tej algebry. Z zadania b) wywnioskować, że algebra ta ma 2^{2^n} elementów.

d) Udowodnić, że każda funkcja dająca się otrzymać ze zmiennej $X_1, \ldots, X_n$ za pomocą skończonej liczby operacji boole'owskich jest równa jakiejś funkcji z B_n.

40. TWIERDZENIA ALGEBRY BOOLE'A

W paragrafie tym podamy kilka ciekawszych twierdzeń, ilustrujących niektóre własności algebry Boole'a.

TWIERDZENIE 1. *W każdej algebrze Boole'a istnieje tylko jeden wyróżniony element* 0 *oraz jeden wyróżniony element* 1.

Dowód. Załóżmy, że twierdzenie 1 jest nieprawdziwe i że istnieją dwa różne elementy 0 i 0* oraz 1 i 1* (tj. $0 \neq 0^*$ oraz $1 \neq 1^*$), spełniające aksjomaty algebry Boole'a dla wszelkich A. Więc w aksjomacie

$$A \cup 0 = A$$

za A możemy podstawić 0*, otrzymując

$$0^* \cup 0 = 0^*. \quad (1)$$

Ponieważ aksjomat A_3 jest również spełniony dla drugiego zera 0^*, więc

$$A \cup 0^* = A.$$

Podstawiając za $A = 0$, otrzymamy

$$0 \cup 0^* = 0. \tag{2}$$

Wobec aksjomatu A_1 z (1) i (2) otrzymamy

$$0 = 0^*,$$

co wykazuje, że oba zera są identyczne.

Postępując podobnie dla jedności otrzymamy, że

$$1 = 1^*.$$

TWIERDZENIE 2. *Dla każdego elementu A dowolnej algebry Boole'a istnieje dokładnie jeden element B taki, że $A \cup B = 1$ i $A \cap B = 0$.*

Z aksjomatów wynika, że istnieje przynajmniej jeden.

Dowód. Załóżmy, że element A ma dwa uzupełnienia A' oraz A^*.

1.	$A^* = A^* \cup 0 =$	$(A_3, 2)$
2.	$= A^* \cup (A \cap A') =$	$(B_4, 3)$
3.	$= (A^* \cup A) \cap (A^* \cup A') =$	$(A_2, 4)$
4.	$= (A \cup A^*) \cap (A^* \cup A') =$	$(A_1, 5)$
5.	$= 1 \cap (A^* \cup A') =$	$(A_4, 6)$
6.	$= 1 \cap (A' \cup A^*) =$	$(A_1, 7)$
7.	$= (A \cup A') \cap (A' \cup A^*) =$	$(A_4, 8)$
8.	$= (A' \cup A) \cap (A' \cup A^*) =$	$(A_2, 9)$
9.	$= A' \cup (A \cap A^*) =$	$(B_4, 10)$
10.	$= A' \cup 0 =$	$(A_3, 11)$
11.	$= A',$	

a więc

$$A' = A^*.$$

TWIERDZENIE 3. *Dla każdego elementu A algebry Boole'a $A'' = A$.*

Dowód. Na podstawie aksjomatów A_1 i B_1, aksjomaty A_4 i B_4 możemy napisać w postaci

$$A' \cup A = 1 \quad \text{oraz} \quad A' \cap A = 0.$$

Wynika stąd, że A jest dopełnieniem A'. Ponieważ na podstawie twierdzenia 2 dopełnienie takie może być tylko jedno, więc

$$A'' = A.$$

TWIERDZENIE 4. *W każdej algebrze Boole'a $1' = 0$ oraz $0' = 1$.*

Dowód. Podstawiając w A_3 oraz B_3 zamiast A element 1, otrzymamy

$$1 \cup 0 = 1, \quad 1 \cap 0 = 0.$$

Na podstawie aksjomatu A_4 i B_4 oraz twierdzenia 2, $0 = 1'$.

Z twierdzenia 3 zaś otrzymamy

$$0' = 1'' = 1.$$

TWIERDZENIE 5. *Dla każdego elementu A algebry Boole'a*

$$A \cup 1 = 1 \quad \textit{oraz} \quad A \cap 0 = 0.$$

Dowód.

1.	$A \cup 1 = (A \cup 1) \cap 1 =$	(B_3)
2.	$= 1 \cap (A \cup 1) =$	$(B_1, 1)$
3.	$= (A \cup A') \cap (A \cup 1) =$	$(A_4, 2)$
4.	$= A \cup (A' \cap 1) =$	$(A_2, 3)$
5.	$= A \cup A' =$	$(B_3, 4)$
6.	$= 1.$	$(A_4, 5)$

Podobnie przebiega dowód dla drugiego przypadku.

1.	$A \cap 0 = (A \cap 0) \cup 0 =$	(A_3)
2.	$= 0 \cup (A \cap 0) =$	$(A_1, 1)$
3.	$= (A \cap A') \cup (A \cap 0) =$	$(B_4, 2)$
4.	$= A \cap (A' \cup 0) =$	$(B_2, 3)$
5.	$= A \cap A' =$	$(A_3, 4)$
6.	$= 0.$	$(B_4, 5)$

TWIERDZENIE 6. *Dla każdego elementu A algebry Boole'a zachodzi:*

$$A \cup A = A \quad \text{oraz} \quad A \cap A = A.$$

Dowód.

1. $A = A \cup 0 =$ (A_3)
2. $= A \cup (A \cap A') =$ (B_4, 1)
3. $= (A \cup A) \cap (A \cup A') =$ (A_2, 2)
4. $= (A \cup A) \cap 1 =$ (A_4, 3)
5. $= A \cup A.$ (B_3, 4)

I podobnie otrzymamy dla drugiego przypadku.

1. $A = A \cap 1 =$ (B_3)
2. $= A \cap (A \cup A') =$ (A_4, 1)
3. $= (A \cap A) \cup (A \cap A') =$ (B_2, 2)
4. $= (A \cap A) \cup 0 =$ (B_4, 3)
5. $= A \cap A.$ (A_3, 4)

TWIERDZENIE 7. *Dla każdych elementów A i B należących do algebry Boole'a zachodzą prawa pochłaniania:*

$$A \cup (A \cap B) = A \quad \textit{oraz} \quad A \cap (A \cup B) = A.$$

Dowód.

1. $A \cup (A \cap B) = (A \cap 1) \cup (A \cap B) =$ (B_3)
2. $= A \cap (1 \cup B) =$ (B_2, 1)
3. $= A \cap (B \cup 1) =$ (A_1, 2)
4. $= A \cap 1 =$ (Tw 5, 3)
5. $= A.$ (B_3, 4)

I podobnie otrzymamy dla drugiego przypadku

1. $A \cap (A \cup B) = (A \cup 0) \cap (A \cup B) =$ (A_3)
2. $= A \cup (0 \cap B) =$ (A_2, 1)
3. $= A \cup (B \cap 0) =$ (B_1, 2)

4. $= A \cup 0 =$ (Tw 5, 3
5. $= A.$ (A_3, 4

Podamy jeszcze bez dowodu dwa twierdzenia.

TWIERDZENIE 8. *Jeżeli A, B i C są elementami algebry Boole'a to*

$$(A \cup B) \cup C = A \cup (B \cup C)$$

oraz

$$(A \cap B) \cap C = A \cap (B \cap C).$$

TWIERDZENIE 9. *Dla każdego A i B należącego do algebry Boole'a zachodzą pewne prawa zwane prawami de Morgana*

$$(A \cap B)' = A' \cup B' \quad \textit{oraz} \quad (A \cup B)' = A' \cap B'.$$

Streszczenie

Podaliśmy przykłady dowodów twierdzeń w algebrach Boole'a. Udowodniliśmy jedyność elementów wyróżnionych i dopełnienia. Podaliśmy prawa dotyczące dopełnienia, prawa łączności „∪" oraz „∩"oraz prawa de Morgana.

Zadania

1. a) Opierając się na udowodnionych w tym paragrafie twierdzeniach i na aksjomatach algebry Boole'a udowodnić, że każda algebra Boole'a jest strukturą rozdzielczą (por. zad. 9 i zad. 14, par. 36, rozdz. V).

b) Co oznacza relacja $A \leqslant B$ w algebrze Boole'a?

c) Udowodnić prawo:

$$(A \cup B) \cap (B \cup C) \cap (C \cup A) = (A \cap B) \cup (B \cap C) \cup (C \cap A)$$

(por. zad. 14, par. 36, rozdz. VI).

2. Udowodnić, że żaden element algebry Boole'a nie może spełniać równania $X = X'$.

3. Udowodnić prawa de Morgana podane w twierdzeniu 9.

4. Udowodnić twierdzenie 8.

§ 41. REPREZENTACJE ALGEBR BOOLE'A

W paragrafie 2 poznaliśmy różne przykłady algebr Boole'a. Wśród nich naczelne miejsce zajmowała algebra zbiorów. Celem rozważań tego paragrafu jest pokazanie, że każdą algebrę Boole'a można traktować jako taką algebrę, której elementami są zbiory.

Rozpatrzmy następujący przykład.

PRZYKŁAD 1. Rozpatrzmy następującą algebrę B złożoną z ośmiu zbiorów. $0 = \{0\}$, $A_1 = \{1, 2\}$, $A_2 = \{3, 4, 5\}$, $A_3 = \{6\}$, $A_4 = \{1, 2, 3, 4, 5\}$, $A_5 = \{3, 4, 5, 6\}$, $A_6 = \{1, 2, 6\}$, $1 = \{1, 2, 3, 4, 5, 6\}$. Łatwo stwierdzić, że jest to algebra Boole'a, jeżeli za operacje przyjmiemy sumę, przekrój i uzupełnienie zbiorów. Rozpatrzmy następujące elementy tej algebry: przyporządkujemy każdemu elementowi A tej algebry zbiór tych spośród elementów A_1, A_2, A_3, które są zawarte (w sensie zawierania zbiorów) w elemencie A. Wtedy elementy przyporządkowane elementowi 0 będą tworzyć zbiór pusty, elementy przyporządkowane elementom A_1, A_2, A_3 utworzą podzbiory jednoelementowe $\{A_1\}, \{A_2\}, \{A_3\}$. Podobnie elementy przyporządkowane A_4, A_5, A_6, utworzą podzbiory dwuelementowe odpowiednio równe $\{A_1, A_2\}, \{A_2, A_3\}, \{A_1, A_3\}$, zbioru $\{A_1, A_2, A_3\}$. przyporządkowanego elementowi 1.

Wypiszmy to przyporządkowanie explicite:

$$0 \leftrightarrow \{0\},$$
$$A_1 \leftrightarrow \{A_1\},$$
$$A_2 \leftrightarrow \{A_2\},$$
$$A_3 \leftrightarrow \{A_3\},$$
$$A_4 \leftrightarrow \{A_1, A_2\},$$
$$A_5 \leftrightarrow \{A_2, A_3\},$$
$$A_6 \leftrightarrow \{A_1, A_3\},$$
$$1 \leftrightarrow \{A_1, A_2, A_3\},$$

Podane przyporządkowanie jest przyporządkowaniem wzajemnie jednoznacznym między elementami algebry B, a wszystkimi podzbiorami zbioru trójelementowego $\{A_1, A_2, A_3\}$.

Można łatwo sprawdzić, że przyporządkowanie to przeprowadza sumę elementów algebry B na sumę odpowiadających im podzbiorów, iloczyn elementów na przekrój odpowiadających im podzbiorów oraz uzupełnienie elementu na uzupełnienie podzbioru. Obie algebry, algebra B i algebra wszystkich podzbiorów zbioru trójelementowego, są więc nie odróżnialne pod względem własności algebraicznych, tzn. takich własności, które dadzą się zapisać za pomocą znaków działań i równości elementów algebry.

Mówi się krótko, że algebry są izomorficzne. Przekształcenie wzajemnie jednoznaczne jednej algebry na drugą zachowujące działania (tzn. przeprowadzające sumę na sumę, iloczyn na iloczyn, uzupełnienie na uzupełnienie) nazywa się *izomorfizmem algebr Boole'a.*

Podany przykład jest częścią ogólniejszego twierdzenia, mówiącego, że *każda skończona algebra Boole'a jest izomorficzna z algebrą wszystkich podzbiorów jakiegoś zbioru skończonego.*

Przed naszkicowaniem dowodu tego twierdzenia wprowadźmy następujące pojęcie.

DEFINICJA 1. Element $A \neq 0$ algebry Boole'a będziemy nazywali *atomem*, jeżeli dla każdego elementu B z tego, że $A \cap B = B$ wynika, że albo $B = 0$ albo $B = A$. Atom A taki, że $A \cap X = X$ będziemy nazywali *atomem elementu X.*

W przykładzie 1, atomami są tylko elementy A_1, A_2, A_3. Na przykład A_4 nie jest atomem, gdyż $A_4 \cap A_1 = A_1$, ale $A_1 \neq 0$ i $A_1 \neq A_4$. Element A_1 jest atomem następujących elementów $A_1, A_4, A_6, 1$.

Z prawa $A \cap A = A$ wynika, że każdy atom A jest atomem siebie samego. Podobnie z prawa $1 \cap A = A$ wynika, że każdy atom A jest atomem elementu 1.

Atomy algebry są takimi jej elementami, że mnożąc je przez dowolny element algebry nie otrzymamy żadnych nowych elementów. W wyniku mnożenia atomu otrzymamy zawsze albo znowu ten sam atom, albo zero algebry.

W przypadku gdy algebra jest skończona atomy są takimi elementarnymi cegiełkami, których przez mnożenie nie można rozdrobnić, sumując zaś je w dowolnych zestawieniach można otrzymać dowolny element algebry.

Zachodzi mianowicie następujące twierdzenie:

TWIERDZENIE 1. *Każdy różny od zera element algebry skończonej ma przynajmniej jeden atom. Każdy element jest sumą (skończoną) swoich atomów.*

Twierdzenie to, którego dowód jest dość żmudny podajemy bez dowodu. Również bez dowodu podamy następujące twierdzenie.

TWIERDZENIE 2. *Przyporządkujmy każdemu elementowi B algebry Boole'a (skończonej lub nie) zbiór $T(B)$ wszystkich jego atomów. Wtedy zachodzą związki*

$$T(B \cup C) = T(B) \cup T(C),$$

$$T(B \cap C) = T(B) \cap T(C),$$

$$T(B') = [T(B)]'.$$

Mówiąc słowami, przyporządkowanie każdemu elementowi algebry Boole'a zbioru jego atomów, przeprowadza sumę elementów algebry na sumę odpowiadających im zbiorów; iloczyn elementów na iloczyn odpowiadających im podzbiorów, uzupełnienie zaś elementu na uzupełnienie zbioru do zbioru wszystkich atomów.

Z twierdzenia 1 można wywnioskować, że jeżeli algebra jest skończona, to przyporządkowanie (opisane w twierdzeniu 2) algebry B na zbiór wszystkich podzbiorów zbioru atomów jest wzajemnie jednoznaczne. Daje to następujące:

TWIERDZENIE 3. *Każda algebra Boole'a skończona jest izomorficzna z algebrą wszystkich podzbiorów jakiegoś zbioru (np. zbioru swoich atomów).*

Otrzymawszy taki wynik wiemy już, że skończone algebry Boole'a pod względem elementarnych własności algebraicznych nie są odróżnialne od algebr wszystkich podzbiorów zbiorów skończonych. Nasuwa się pytanie czy zachodzi twierdzenie ogólniejsze, że każda algebra Boole'a nawet nieskończona jest izomorficzna z algebrą wszystkich podzbiorów jakiegoś zbioru (być może nieskończonego). Stosunkowo nietrudno pokazać, że takie twierdzenie nie może zachodzić.

Prawdziwe jest natomiast twierdzenie następujące:

TWIERDZENIE 4. *Każda algebra Boole'a jest izomorficzna z algebrą Boole'a pewnych (być może nie wszystkich) podzbiorów jakiegoś zbioru.*

Twierdzenie to nazywa się *twierdzeniem o reprezentacji algebr Boole'a*, lub *twierdzeniem Stone'a*. Dowód, z uwagi na metody jakich trzeba w nim użyć, korzystające z mocnych twierdzeń teorii mnogości, wykracza poza ramy tej książki. Odsyłamy więc czytelnika do obszerniejszych książek: Birkhoffa lub Sikorskiego.

Streszczenie

Naszkicowaliśmy dowód twierdzenia, że każda algebra Boole'a skończona jest izomorficzna z algebrą wszystkich podzbiorów jakiegoś zbioru skończonego. Zauważyliśmy, że prawdziwe jest twierdzenia ogólniejsze, że każda algebra Boole'a jest izomorficzna z algebrą jakichś podzbiorów (może nie wszystkich) jakiegoś zbioru nieskończonego.

Zadania

1. Udowodnić, że jeżeli 1 jest atomem algebry, to algebra jest dwuelementowa.

2. Udowodnić, że każdy element algebry Boole'a skończonej, tzn. zawierającej skończoną liczbę elementów, ma atom.

Wskazówka. Niech $A_1, \ldots, A_m$ będą wszystkimi elementami algebry różnymi od 0 i 1. Jeżeli B jest elementem algebry różnym od zera, to określamy ciąg elementów $B_1, \ldots, B_m$ następująco,

$$B_0 = B,$$

$$B_{i+1} = \begin{cases} B_i \cap A_{i+1}, & \text{gdy} \quad B_i \cap A_{i+1} \neq 0, \\ B_i, & \text{gdy} \quad B_i \cap A_{i+1} = 0. \end{cases}$$

Element $A = B_m$ będzie atomem elementu B.

3. Udowodnić, że w algebrze skończonej każdy element jest sumą swoich atomów.

4. a) Udowodnić, że dla odwzorowania T z twierdzenia 2, zachodzą związki:

$$T(B) \cup T(C) \subset T(B \cup C),$$

$$T(B) \cap T(C) \subset T(B \cap C),$$

$$T(B) \cup T(B') = \text{zbiór wszystkich atomów algebry},$$

$$T(B) \cap T(B') = \varnothing.$$

b) W oparciu o powyższe udowodnić:

$$T(A) \cap T(B) = T(A \cap B).$$

c) Udowodnić, że

$$[T(B)]' = T(B').$$

d) Z b) i c) wywnioskować, że

$$T(B) \cup T(C) = T(B \cup C).$$

5. Udowodnić, że jeśli algebra jest skończona to:

a) Opisane w twierdzeniu 2 odwzorowanie przeprowadza różne elementy na różne, tzn. jeżeli $B \neq C$, to $T(B) \neq T(C)$.

Wskazówka. Jeżeli $B \neq C$, to istnieje atom należący do jednego elementu, a nie należący do drugiego.

b) Dla każdego podzbioru S zbioru atomów istnieje element D taki, że $T(D) = S$.

Wskazówka. Wziąć za D sumę, w sensie $A_1 \cup A_2 \cup \ldots \cup A_k$, wszystkich atomów $A_1, \ldots, A_k$ należących do D.

6. Z twierdzenia 3 wywnioskować, że każda algebra Boole'a skończona ma liczbę elementów postaci 2^n.

7. Udowodnić, że rachunek zdań w przypadku, gdy liczba zmiennych zdaniowych jest nieskończona, traktowany jako algebra Boole'a, tak jak w przykładzie 2, z paragrafu 38, jest algebrą Boole'a, w której żaden element nie ma atomu. Algebry takie nazywamy *algebrami bezatomowymi*.

§ 42. ZNACZENIE TWIERDZEŃ O REPREZENTACJI

Omówimy teraz wnioski jakie płyną z twierdzenia o reprezentacji algebr Boole'a.

Algebry Boole'a wprowadziliśmy aksjomatycznie w § 38 tego rozdziału, używając języka potocznego. Teorię tę można również przedstawić w postaci elementarnej teorii sformalizowanej (zob. r. IV).

Formułami atomowymi są formuły postaci równości dwóch termów.

(1) $$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$$

Termy są to wyrażenia zbudowane w omówiony sposób ze zmiennych $x_1, x_2, x_3, \ldots$ stałych 0 oraz 1 (działań zeroargumentowych), symboli „′" działania jednoargumentowego, oraz symboli „∪" i „∩" działań dwuargumentowych. Na przykład wyrażenia;

$$\big((X_1 \cap X_3) \cup (X_2')\big)' \cup (0 \cup X_1),$$

$$X_1 \cup (X_2 \cup X_3), \quad 1' \cup (0 \cap 0), \quad X_5, \quad X_1', \quad 0, \quad 1$$

są termami, natomiast wyrażenie:

$$X_1 \cap \cap' (X_1 \cup 0)$$

nie jest termem, gdyż nie jest poprawnie zbudowane.

Z formuł postaci (1) budujemy inne formuły, łącząc je spójnikami zdaniowymi i opatrując kwantyfikatorami.

Formułą będzie na przykład

$$\mathrm{A}x_1\, \mathrm{E}x_2\, (x_1 \cup x_2 = x_2)\ \&\ {\sim}(0 = x_1),$$

czy też

$$\mathrm{A}x_1\, (x_1 = x_1).$$

Formuła ta jest zdaniem, gdyż nie zawiera zmiennych wolnych. Lecz już formuła

$$\mathrm{E}x_2\, [(x_1 \cup x_1 = x_2)\ \&\ (0 = x_1)],$$

czy też formuła

$$x_1 = x_1$$

nie są zdaniami, gdyż zawierają zmienną wolną x_1.

Aksjomatami teorii będą następujące zdania:

1. Aksjomat dotyczący elementów wyróżnionych

$${\sim}(0 = 1).$$

2. Aksjomaty równości:

$$\mathrm{A}x_1\, (x_1 = x_1),$$

$$\mathrm{A}x_1\, \mathrm{A}x_2\, [(x_1 = x_2) \Rightarrow (x_2 = x_1)],$$

$$\mathrm{A}x_1\, \mathrm{A}x_2\, \mathrm{A}x_3\, [(x_1 = x_2)\ \&\ (x_2 = x_3) \Rightarrow (x_1 = x_3)],$$

$$\mathrm{A}x_1\, \mathrm{A}x_2\, \mathrm{A}x_3\, [(x_1 = x_2) \Rightarrow \{(x_1 \cup x_3 = x_2 \cup x_3)\ \&\ (x_1 \cap x_3 = x_2 \cap x_3)\ \&\ (x_1' = x_2')\}].$$

3. Aksjomaty algebry Boole'a:

$$\mathrm{A}x_1\, \mathrm{A}x_2\, (x_1 \cup x_2 = x_2 \cup x_1), \quad \mathrm{A}x_1\, \mathrm{A}x_2\, (x_1 \cap x_2 = x_2 \cap x_1),$$

$$\mathrm{A}x_1\, \mathrm{A}x_2\, \mathrm{A}x_3\, \big(x_1 \cup (x_2 \cap x_3) = (x_1 \cup x_2) \cap (x_1 \cup x_3)\big),$$

$$Ax_1\, Ax_2\, Ax_3\, \big(x_1 \cap (x_2 \cup x_3) = (x_1 \cap x_2) \cup (x_1 \cap x_3)\big)$$

$$Ax_1\,(x_1 \cup 0 = x_1), \quad Ax_1\,(x_1 \cap 1 = x_1),$$

$$Ax_1\,(x_1 \cup x_1' = 1), \quad Ax_1\,(x_1 \cap x_1' = 0).$$

Należy zwrócić uwagę na różnicę między aksjomatami podanymi tutaj (aksjomatami teorii) a aksjomatami algebry Boole'a, (aksjomatami algebry) podanymi w § 38. Tamte nie były zdaniami lecz pewnego rodzaju schematami zdań, w których za litery A, B, C można było wstawiać dowolne elementy algebry Boole'a (tzn. zbioru X) i otrzymywać konkretne związki (zdania mówiące o tych związkach) między konkretnymi działaniami określające własności działań „$\cup$", „$\cap$", „$'$" na elementach zbioru X.

Aksjomaty algebry Boole'a, jako elementarnej teorii sformalizowanej, są pewnymi zdaniami, które a priori przyjmujemy za twierdzenia teorii i posługując się regułami wnioskowania oraz aksjomatami logicznymi otrzymujemy nowe twierdzenia teorii — konsekwencje aksjomatów.

Teoria algebr Boole'a jest niesprzeczna. Ma przynajmniej jeden model — algebrę dwuelementową, w której zbiór $X = \{0, 1\}$.

Badanie teorii jest o tyle wartościowe i ciekawe, że każde twierdzenie teorii jest prawdziwe w każdym jej modelu. Teoria algebr Boole'a jak każda teoria elementarna ma nieskończenie wiele różnych modeli, nieizomorficznych ze sobą i bardzo ważnych ze względu na zastosowania, np.: rachunek zbiorów, sieci logiczne, oraz inne modele podane w przykładach.

Twierdzenie o reprezentacji algebr Boole'a mówi, że każdego modelu można szukać, z dokładnością do izometrii, w algebrze pewnych podzbiorów ustalonego zbioru. Znaczy to, że zmienne $x_1, x_2, \ldots$ można uważać za (interpretować jako) pewne podzbiory jakiegoś zbioru, stałe 0 i 1 za zbiór pusty i zbiór pełny, a symbole działań „$\cup$", „$\cap$", „$'$" za znaki działań: sumowania, przekroju i dopełnienia zbiorów.

Streszczenie

Przedstawiliśmy algebry Boole'a w postaci elementarnej teorii sformalizowanej. Twierdzenie o reprezentacji pozwala szukać modeli tej teorii w algebrze zbiorów.

Zadania

1. Udowodnić, że zbiór termów tworzy algebrę Boole'a ze względu na operacje łączenia termów znakiem „$\cup$", „$\cap$" oraz stawiania znaku „$'$" po termie.

Uwaga. Termy stanowiące różne ciągi symboli uważamy za różne.

2. Rozpisać aksjomaty A_1—B_4 z § 37 na poszczególne zdania, w przypadku gdy zbiór X jest dwuelementowy $X = \{0, 1\}$. Na przykład $A \cup A = A$ rozpisujemy jako $0 \cup 0 = 0$ i $1 \cup 1 = 1$.

3. Udowodnić, że jeżeli zdanie

$$Ax_1 \ldots Ax_n (f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n))$$

mające postać równości dwóch termów opatrzonych kwantyfikatorem ogólnym jest prawdziwe dla wszystkich podstawień za zmienne x_i w termie elementów 0 i 1 z algebry dwuelementowej $X = \{0, 1\}$, to jest twierdzeniem teorii (Wskazówka. Udowodnić, że jest prawdziwe w każdym modelu).

Rozdział 8

TEORIA PÓŁGRUP

Teoria półgrup stanowi ciekawy przykład przejścia od badania konkretnych faktów matematycznych — własności zbiorów przekształceń, do teorii aksjomatycznej opisującej te własności. Zajmuje się ona właściwie badaniem własności składania przekształceń, zakładając o nich tylko tyle, że są zawsze wykonalne i łączne. Aksjomatyka teorii półgrup jest więc bardzo uboga, zawiera jeden aksjomat będący równością wyrażającą prawo łączności. Ubóstwo założeń powoduje siłą rzeczy ubóstwo wniosków i ich ogólność.

Zdawać by się mogło, że przy tak skromnych założeniach teoria półgrup nie może stanowić interesującego obiektu badań. Tak jednak nie jest. Bogactwo modeli czyni tę teorię interesującą i nadającą się do zastosowań w wielu różnych dziedzinach.

Matematyka zainteresuje teoria półgrup głównie dzięki różnym wzmocnieniom jej aksjomatyki, które wiążą się z pojęciem grupy i pojęciem pierścienia rozpatrywanym w algebrze. Będąc w zgodzie z historią rozwoju tych dyscyplin należałoby powiedzieć, że teoria półgrup może ciekawić matematyka jako uogólnienie teorii grup i teorii pierścieni.

Logika i badacza podstaw matematyki zainteresuje teoria półgrup, jako teoria stosunkowo prosta, lecz już dostatecznie bogata, by mieć interesujące własności — jako prosta ilustracja twierdzeń z podstaw matematyki.

Powodem zamieszczenia w tej książce wiadomości o teorii półgrup są jej zastosowania do opisu automatów skończonych, oraz brak przystępnych wprowadzeń w teorię półgrup. Od strony algebraicznej najpełniej wyłożona jest teoria półgrup w monografii Lapina. Pewne wiadomości można znaleźć również w książce Kurosza [1965]. Z uwagi na brak odpowiednio przystępnej literatury z teorii półgrup, oprócz definicji i wyjaśnienia zna-

czenia pojęcia półgrupy (§§ 43-45) poświęciliśmy dwa paragrafy (§§ 46-47) omówieniu bardziej specjalnych pojęć algebraicznych: homomorfizmu i kongruencji. Paragrafy te są przydatne do zrozumienia § 48, poświęconego określaniu półgrupy przez równości określające i najbardziej może pod względem logicznym interesującego § 49 poświęconego zagadnieniom rozstrzygalności w teorii półgrup.

§ 43. WPROWADZENIE

Aby zrozumieć skąd wzięła się podana w następnym paragrafie definicja półgrupy, rozpatrzmy kilka przykładów zbiorów, w których określona jest operacja „składania" elementów.

PRZYKŁAD 1. Rozpatrzmy zbiór $\mathbf{M}_n$ macierzy kwadratowych ustalonego stopnia np. Jeżeli

$$\mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \quad \text{i} \quad \mathbf{B}=\begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix}$$

są dwoma takimi macierzami, to macierz

$$\mathbf{C}=\begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & \dots & c_{nn} \end{bmatrix},$$

gdzie

$$c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+\dots+a_{in}b_{nj}=\sum_{k=1}^{n}a_{ik}b_{kj},$$

również należącą do zbioru $\mathbf{M}_n$, nazywamy *iloczynem* lub *złożeniem macierzy* **A** i **B**, co zapisujemy

$$\mathbf{C}=\mathbf{A}\cdot\mathbf{B}.$$

Łatwo można stwierdzić, że mnożenie macierzy jest łączne, tzn. spełnia prawo

$$\mathbf{A}\cdot(\mathbf{B}\cdot\mathbf{C})=(\mathbf{A}\cdot\mathbf{B})\cdot\mathbf{C},$$

gdzie **A**, **B** i **C** są dowolnymi macierzami ze zbioru $\mathbf{M}_n$ (tzn. macierzami kwadratowymi stopnia n).

Mnożenie macierzy ma tylko niektóre własności podobne do mnożenia liczb. Na przykład łatwo można sprawdzić, że macierz

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ 0 & 0 & 1 \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 \dots & 1 \end{bmatrix} \quad \text{(jedynki na głównej przekątnej)}$$

gra rolę jedności, tzn. dla każdej macierzy $\mathbf{A}$ ze zbioru $\mathbf{M}_n$ spełnione jest prawo

$$\mathbf{A}\cdot\mathbf{E} = \mathbf{E}\cdot\mathbf{A} = \mathbf{A}.$$

Podobnie macierz

$$\mathbf{O} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 \end{bmatrix} \quad \text{(same zera)}$$

gra rolę zera, tzn. spełnia prawo:

$$\mathbf{O}\cdot\mathbf{A} = \mathbf{A}\cdot\mathbf{O} = \mathbf{O},$$

dla każdej macierzy $\mathbf{A}$ ze zbioru $\mathbf{M}_n$.

Dalszych analogii z mnożeniem liczb jednak nie ma.

Dla mnożenia liczb zachodzi prawo przemienności:

$$a\cdot b = b\cdot a.$$

Nie zachodzi jednak prawo przemienności dla mnożenia macierzy, tzn. iloczyny $\mathbf{A}\cdot\mathbf{B}$ i $\mathbf{B}\cdot\mathbf{A}$ są na ogół różne. Na przykład dla macierzy $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ i $\mathbf{B} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$ ze zbioru $\mathbf{M}_2$ mamy

$$\mathbf{A}\cdot\mathbf{B} = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}\cdot\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix},$$

zaś

$$\mathbf{B}\cdot\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}\cdot\begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 3 & 5 \\ 0 & 0 \end{bmatrix},$$

a więc w tym przypadku $\mathbf{A}\cdot\mathbf{B} \neq \mathbf{B}\cdot\mathbf{A}$.

Dla liczb zachodzi prawo skracania:

Jeżeli $a\cdot b = a\cdot c$, to gdy $a \neq 0$ musi być $b = c$.

Prawo to nie zachodzi dla mnożenia macierzy. Na przykład gdy

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

to wobec

$$\mathbf{A}\cdot\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}\cdot\begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\mathbf{A}\cdot\mathbf{C} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

mamy $\mathbf{A}\cdot\mathbf{B} = \mathbf{A}\cdot\mathbf{C}$, ale $\mathbf{B} \neq \mathbf{C}$, choć $\mathbf{A} \neq \mathbf{O}$.

PRZYKŁAD 2. Podobnie sprawa wygląda, gdy rozpatrujemy składanie przekształceń jakiegoś zbioru w siebie. Niech S będzie zbiorem. Dla większej poglądowości przyjmijmy, że S jest zbiorem skończonym, składającym się z elementów $a_1, \dots, a_n$. Funkcję F, która każdemu elementowi z S przyporządkowuje znowu element z S nazywamy *przekształceniem zbioru S w siebie*(1).

Jeżeli dwa przekształcenie F i G opiszemy za pomocą tabelek:

$$F: \begin{array}{l} a_1 \to F(a_1) \\ a_2 \to F(a_2) \\ \dots\dots\dots \\ a_n \to F(a_n) \end{array} \qquad G: \begin{array}{l} a_1 \to G(a_1) \\ a_2 \to G(a_2) \\ \dots\dots\dots \\ a_n \to G(a_n) \end{array}$$

(1) Słowo „na" oznacza więc odwzorowanie na cały zbiór, a „w" oznacza odwzorowanie w część. Inaczej mówiąc przekształcenie F zbioru S jest przekształceniem w zbiór S, jeżeli wszystkie wartości $F(x)$ dla $x \in S$ należą do S. Nie żądamy przy tym by

(*) dla każdego $y \in S$ istniało takie $x \in S$, że $F(x) = y$.

Przekształcenie F spełniające warunek (*) nazywamy przekształceniem zbioru S na zbiór S. Klasa przekształceń „na" jest więc węższą klasą niż klasa przekształceń „w".

to złożenie H przekształceń F i G opisujemy za pomocą tabelki

$$H = G \cdot F \quad \begin{array}{l} a_1 \to G(F(a_1)) \\ a_2 \to G(F(a_2)) \\ \cdots\cdots\cdots \\ a_n \to G(F(a_n)) \end{array}$$

Jest to znowu przekształcenie zbioru S w siebie. W tym przypadku mówimy, że zbiór $P(S)$ przekształceń zbioru S w siebie jest zamknięty względem składania przekształceń. Łatwo zauważyć, że składanie przekształceń jest łączne, tzn. że

$$F \cdot (G \cdot H) = (F \cdot G) \cdot H$$

dla każdych przekształceń $F, G, H \in P(S)$. Rolę jedności gra przekształcenie tożsamościowe, tzn. przekształcenie E opisane za pomocą tabelki

$$E \quad \begin{array}{l} a_1 \to E(a_1) = a_1 \\ a_2 \to E(a_2) = a_2 \\ \cdots\cdots\cdots \\ a_n \to E(a_n) = a_n \end{array}$$

Rzeczywiście łatwo zauważyć, że

$$E \cdot F = F \cdot E = F, \quad \text{dla każdego } F \in P(S).$$

Można udowodnić, że w zbiorze $P(S)$ nie ma elementu 0 takiego, że

$$F \cdot 0 = 0 \cdot F = 0, \quad \text{dla każdego } F \in P(S).$$

Można również pokazać, że nie zachodzą prawa skracania:

$$F \cdot G = F \cdot H \quad \text{wynika} \quad G = H$$

(*prawo lewostronnego skracania*),

$$G \cdot F = H \cdot F \quad \text{wynika} \quad G = H$$

(*prawo prawostronnego skracania*).

PRZYKŁAD 3. Działanie mnożenia określone i wykonalne w zbiorze N liczb naturalnych 0, 1, 2, 3, ... spełnia większą ilość praw, niż działanie mnożenia macierzy określone w zbiorze $\mathbf{M}_n$, czy też działanie składania

przekształceń określone w zbiorze $P(S)$. Dla mnożenia liczb naturalnych zachodzi zarówno prawo łączności

$$n \cdot (m \cdot r) = (n \cdot m) \cdot r, \quad \text{dla } n, m, r, \in N,$$

jak i prawo przemienności

$$n \cdot m = m \cdot n, \quad \text{dla } n, m \in N$$

oraz prawa skracania przez elementy $\neq 0$. Dzięki przemienności wystarczy podać jedno z nich np. lewostronne prawo skracania:

$$\text{jeżeli } n \cdot m = n \cdot r \text{ i } n \neq 0, \text{ to } m = r.$$

Jednością jest liczba 1, gdyż

$$1 \cdot n = n \cdot 1 \quad \text{dla każdego } n \in N.$$

zerem zaś liczba zero, gdyż

$$0 \cdot n = n \cdot 0 = 0 \quad \text{dla każdego } n \in N.$$

Stąd np. $0 \cdot n = 0(n+1)$, ale ponieważ $n \neq n+1$, więc prawo skracania przez 0 nie zachodzi.

Streszczenie

Opisaliśmy mnożenie macierzy i składanie przekształceń. Zwróciliśmy uwagę na prawa dotyczące tych operacji, w szczególności na prawo łączności dotyczące tych działań. Zbadaliśmy własności zera i jedynki oraz omówiliśmy prawa skracania.

Zadania

1. Podać przykład, że dla mnożenia macierzy stopnia n, nie zachodzi prawo skracania prawostronnego, tzn. że z tego, że $\mathbf{B} \cdot \mathbf{A} = \mathbf{C} \cdot \mathbf{A}$ i $\mathbf{A} \neq \mathbf{0}$, nie można wnioskować, że $\mathbf{B} = \mathbf{C}$.

2. Opisać tabelkami wszystkie $3^3 = 27$ przekształceń zbioru $S = \{a_1, a_2, a_3\}$ w siebie.

a) Udowodnić, że składanie przekształceń jest łączne.

b) Podać przykład, że składanie to jest nieprzemienne.

c) Jakie przekształcenie jest jednością półgrupy $P(S)$?

d) Czy zbiór $P(S)$ ma zero?

e) Czy zachodzą lewo i prawostronne prawa skracania?

3. Opisać za pomocą tabelek wszystkie $2^2 = 4$ przekształcenia zbioru $S = \{a_1, a_2\}$ składającego się z dwóch elementów. Oznaczyć te przekształcenia symbolami $F_1, \ldots, F_4$ i opisać tabelkę składania w $P(S)$ według wzoru:

	F_1	F_2	F_3	F_4
F_1	$F_1 \cdot F_1$	$F_1 \cdot F_2$	...	...
F_2	$F_2 \cdot F_1$	...	...	...
F_3	...	...	...	...
F_4	...	...	...	$F_4 \cdot F_4$

Wskazać jedność zbioru $P(S)$ oraz idempotenty, tzn. takie elementy f, że $f \cdot f = f$.

4. Udowodnić, że jeżeli zbiór S ma n elementów, to zbiór $P(S)$ składa się z n^n przekształceń.

§ 44. DEFINICJA PÓŁGRUPY

W tym paragrafie podamy definicję półgrupy.

DEFINICJA. *Półgrupą* nazywamy zbiór P dowolnych elementów, jeżeli spełnione są następujące warunki.

1. W zbiorze P określona jest operacja (działanie) dwuargumentowe, zwana składaniem, przyporządkowująca każdym dwom elementom U i V ze zbioru P, element $W = U \cdot V$ ze zbioru P.

2. Na elementach zbioru P określona jest relacja równości, oznaczona przez „=". Relacja taka spełnia oczywiście następujący warunek:

$$\text{jeżeli } U = U_1 \text{ i } V = V_1, \text{ to } UV = U_1 V_1.$$

3. Operacja wymieniona w punkcie 1 spełnia następujący aksjomat:

P_1. $U \cdot (V \cdot S) = (U \cdot V) \cdot S$ dla każdych $U, V, S \in P$,

zwany *prawem łączności*.

Podane w poprzednim paragrafie przykłady są więc przykładami półgrup, gdy relacja „=" jest zwykłą relacją równości, odpowiednio: macierzy, funkcji oraz liczb całkowitych, składaniem zaś jest odpowiednio: mnożenie macierzy, składanie funkcji oraz mnożenie liczb całkowitych.

Rozpatrzmy teraz pewne specyficzne przypadki półgrup.

Półgrupa nazywa się *przemienną*, jeżeli spełniony jest dodatkowo następujący aksjomat:

P_2. $U \cdot V = V \cdot U$ dla każdych $U, V \in P$,

zwany *prawem przemienności*.

Gdy zachodzi aksjomat:

P_3. Jeżeli $X \cdot U = X \cdot V$, to $U = V$ dla dowolnych $X, U, V \in P$,

zwany *lewostronnym prawem skracania*, wtedy półgrupa nazywa się *półgrupą z lewostronnym skracaniem*. Podobnie jeżeli zachodzi aksjomat zwany *prawostronnym prawem skracania*

P_4. Jeżeli $U \cdot X = V \cdot X$, to $U = V$ dla dowolnych $U, V, X \in P$,

to półgrupa nazywa się *półgrupą z prawostronnym skracaniem*.

Jeżeli zachodzą oba prawa skracania P_3 i P_4 to półgrupa nazywa się *półgrupą ze skracaniem* (por. zadanie 1).

We wszystkich trzech przykładach podanych poprzednio nie zachodziło żadne z praw skracania. Dla pierwszych dwóch przykładów nie zachodziło prawo przemienności, półgrupa zaś podana w przykładzie trzecim jest przemienna.

Teoria półgrup P jest teorię aksjomatyczną mającą jeden aksjomat P_1. Dodając do tego aksjomatu którykolwiek aksjomat P_2, P_3, P_4 otrzymujemy rozszerzenia teorii półgrup:

PK teorie półgrup przemiennych (aksjomaty P_1 i P_2),

PL teorie półgrup z lewostronnym skracaniem (aksjomaty P_1 i P_3),

PP teorie półgrup z prawostronnym skracaniem (aksjomaty P_1 i P_4).

Twierdzenia teorii PL i PP wspólnie tworzą teorię PS półgrup ze skracaniem.

Teorie te są bardzo ubogie, zawierają mało pojęć pierwotnych i mało aksjomatów. Twierdzenia tych teorii są jednak bardzo ogólne.

Podamy teraz przykłady kilku twierdzeń z teorii półgrup.

DEFINICJA 1. Element E należący do półgrupy P nazywamy *jednością półgrupy* P, jeżeli $E \cdot A = A \cdot E = A$ dla każdego $A \in P$.

TWIERDZENIE 1. *W półgrupie istnieje co najwyżej jedna jedność.*

Dowód. Trzeba udowodnić, że jeżeli

(1) $AE = EA = A$ dla każdego $A \in P$,

(2) $BJ = JB = B$ dla każdego $B \in P$,

to

$$E = J.$$

Z (1) wynika, że $JE = J$ (kładąc $A = J$), z (2) zaś wynika, że $JE = E$ (kładąc $B = E$). Stąd żądana teza $J = E$.

DEFINICJA 2. Element 0 półgrupy P nazywamy jej *zerem*, jeżeli $0 \cdot A = A \cdot 0 = 0$ dla każdego A z półgrupy P.

TWIERDZENIE 2. *Półgrupa może mieć co najwyżej jedno zero.*

Jeżeli półgrupa jest półgrupą ze skracaniem (którymkolwiek), to nie może mieć zera. Prawdziwe jest bowiem twierdzenie.

TWIERDZENIE 3. *Jeżeli półgrupa mająca co najmniej dwa elementy ma zero, to nie spełnia żadnego z aksjomatów* P_3 i P_4 (por. zad. 5d i 7).

Teoria półgrup staje się ciekawym obiektem badań matematyki, jeżeli wprowadzamy do niej działania i stałe oraz nowe aksjomaty.

Rozważmy następujące dwa aksjomaty:

P_5. dla każdych A i $B \in P$ istnieje X takie, że $X \cdot A = B$;

P_6. dla każdych A i $B \in P$ istnieje takie Y, że $A \cdot Y = B$,

zwane odpowiednio *aksjomatami istnienia lewostronnej i prawostronnej odwrotności.*

Półgrupa dla której każdy element ma lewą i prawą odwrotność nazywa się *grupą*, teoria G zaś oparta o aksjomaty P_1, P_5 i P_6 nazywa się *teorią grup*. Badanie własności teorii grup stanowi ważną gałąź matematyki, poświęcone jej są liczne monografie (patrz Kurosz [1967] lub M. Hall).

Interesującą teorię można otrzymać dołączając do teorii półgrup nowe działanie „+" i następujące aksjomaty:

$$A+(B+C) = (A+B)+C,$$

$$A+B = B+A,$$

dla każdych A i B istnieje Z takie, że

$$A+Z = B,$$

$$A \cdot (B+C) = (A \cdot B)+(A \cdot C).$$

Teoria taka nazywa się *teorią pierścieni* i stanowi ważny obiekt badań algebry. Z elementami teorii grup i teorii pierścieni zapoznać się może czytelnik z książki A. Mostowskiego i M. Starka.

Streszczenie

Podaliśmy definicję półgrupy przemiennej, półgrupy ze skracaniem lewym (lub prawym). Zdefiniowaliśmy zero i jedność półgrupy.

Zadania

1. Uzasadnić, że zbiór N^* liczb naturalnych dodatnich (bez zera), z działaniem mnożenia i zwykłą relacją równości stanowi przykład półgrupy przemiennej ze skracaniem i jednością, ale bez zera.

2. Podać przykłady półgrupy, w której zachodzi lewostronne prawo skracania (aksjomat P_3), a nie zachodzi prawostronne prawo skracania.

3. Udowodnić, że jeżeli działanie składania w zbiorze dwuelementowym $\{X, Y\}$ określimy jak następuje: $Y \cdot Y = X$, $X \cdot X = Y$, $X \cdot Y = Y$, $Y \cdot X = X$, to prawo łączności nie zachodzi. Zbiór ten nie jest więc półgrupą.

4. Z jakich praw relacji i rachunku kwantyfikatorów korzystaliśmy w dowodzie twierdzenia 1?

Zapisać dowód twierdzenia 1 w postaci sformalizowanej.

5. Podać przykład półgrupy:

a) nie mającej ani zera ani jedynki;

b) mającej jedność i nie mającej zera;

c) mającej zero i nie mającej jedności;

d) podać przykład półgrupy w której jedność jest zarazem zerem.

Ile elementów musi mieć taka półgrupa?

6. a) Udowodnić twierdzenie 2.

b) Zapisać twierdzenie 2 w postaci sformalizowanej.

7. Udowodnić twierdzenie 3.

8. Rozpatrzmy teorię zawierającą relację „=" równości, działanie dwuargumentowe $C = A \cdot B$, działanie jednoargumentowe $B = A^{-1}$ i stałą E (działanie jednoargumentowe) spełniające aksjomaty

G_1. $A \cdot (B \cdot C) = (A \cdot B) \cdot C$,

G_2. $A \cdot E = E \cdot A = A$,

G_3. $A \cdot A^{-1} = A^{-1} \cdot A = E$.

Udowodnić, że działanie musi wtedy spełniać aksjomaty P_1, P_5 i P_6. Określić w teorii o aksjomatach P_1, P_5 i P_6 (teorii grup) określonej w tekście, stałą E i działanie A^{-1} tak by spełnione były aksjomaty G_1, G_2, G_3.

9. Opisać tabelkami wszystkie nieizomorficzne półgrupy czteroelementowe. Która z nich jest izomorficzna z półgrupą z zadania 3 z poprzedniego paragrafu?

§ 45. PÓŁGRUPY PRZEKSZTAŁCEŃ

Celem tego paragrafu jest dowód twierdzenia, że każdą półgrupę można uważać za pewną półgrupę przekształceń.

Weźmy jakiś zbiór S, zawierający przynajmniej jeden element. Jak wiemy z rozważań paragrafu 43, zbiór $P(S)$ wszystkich przekształceń zbioru S w siebie, tworzy półgrupę, z operacją składania przekształceń jako operacją mnożenia w półgrupie. Półgrupa ta jest półgrupą z jednością. Jednością jest przekształcenie tożsamościowe E zbioru S.

Weźmy teraz dowolną półgrupę P, z jednością. Czy można uważać półgrupę P za półgrupę przekształceń, być może nie wszystkich a tylko niektórych, jakiegoś zbioru S w siebie?

Niech elementami półgrupy P będą

$$E,\ A,\ B,\ C,\ \ldots,\ V,\ \ldots \tag{1}$$

gdzie E jest jednością. Pomnożmy te elementy z lewej strony przez jakiś ustalony element V półgrupy P. Spowoduje to pewne przekształcenia zbioru (1) w siebie. Elementy jego przejdą odpowiednio na elementy również należące do (1):

$$V\cdot E = V,\quad V\cdot A,\quad V\cdot B,\quad V\cdot C,\quad \ldots,\quad V\cdot V,\quad \ldots \tag{2}$$

Mnożenie z lewej strony przez V, elementów zbioru (1) wyznacza więc przekształcenie F_V, takie, że

$$E \to F_V(E) = V\cdot E = V,$$
$$A \to F_V(A) = V\cdot A,$$
$$B \to F_V(B) = V\cdot B,$$
$$\ldots\ldots\ldots\ldots\ldots$$

Element U różny od V wyznacza inne przekształcenie F_U zbioru (1) w siebie

$$E \to F_U(E) = U\cdot E = U,$$
$$A \to F_U(A) = U\cdot A,$$
$$B \to F_U(B) = U\cdot B.$$

Przekształcenia F_V i F_U są różne, gdyż wobec

$$F_V(E) = V,\quad F_U(E) = U$$

przekształcają jeden i ten sam element E na różne elementy.

Rozpatrzmy teraz jak wygląda złożenie $F_V\cdot F_U$ przekształceń F_U i F_V czyli takie przekształcenie H, że $H(X) = F_V(F_U(X))$ dla każdego X ze zbioru (1). Czy odpowiada ono pomnożeniu elementów zbioru (1) przez jakiś element W półgrupy P, ze strony lewej?

Ponieważ $F_U(X) = U\cdot X$ więc $F_V(F_U(X)) = F_V(U\cdot X) = V\cdot(U\cdot X) = (V\cdot U)\cdot X$. Wykonanie tego przekształcenia $F_V\cdot F_U$ na elementach zbioru (1) odpowiada pomnożeniu elementów tego zbioru przez element $V\cdot U$. Przekształcenie H jest więc przekształceniem $F_{V\cdot U}$. Wynika stąd, że zbiór F tych przekształceń zbioru (1) w siebie które są postaci $F_U(X) = U\cdot X$, gdzie U jest elementem półgrupy P, jest zamknięty ze względu na składanie, tzn. złożenie przekształceń ze zbioru F jest znowu przekształceniem ze zbioru F. Ponieważ składanie przekształceń jest zawsze łączne, więc omawiany zbiór F przekształceń tworzy półgrupę z operacją składania jako operacją mnożenia.

Przyporządkowanie każdemu elementowi U z półgrupy P przekształcenia F_U z półgrupy F jest jak już zauważyliśmy wzajemnie jednoznacznym odwzorowaniem P na F. Odwzorowanie to wobec wzoru

$$F_U\cdot F_V = F_{U\cdot V}$$

przeprowadza iloczyn elementów półgrupy P, na iloczyn (złożenie elementów) przekształceń półgrupy F. Wyrażamy to krótko mówiąc, że przekształcenie przyporządkowujące elementom $V\in P$ element $F_V\in F$ jest izomorfizmem półgrupy P z półgrupą F. Możemy to wypowiedzieć jako twierdzenie. (1)

TWIERDZENIE 4. *Każda półgrupa z jednością jest izomorficzna z półgrupą pewnych (niekoniecznie wszystkich) przekształceń jakiegoś zbioru w siebie.*

Twierdzenie to stanowi ważny wynik dotyczący półgrup. Mówi ono, że półgrupy z jednością można uważać za półgrupy pewnych przekształceń, gdyż elementowi $V\in P$ odpowiada przekształcenie F_V z F. Składaniu elementów w P odpowiada składanie odpowiednich przekształceń w F. Badania takich półgrup to w istocie badanie półgrup przekształceń. Tłumaczy to w pewnym stopniu ważność i zastosowanie teorii półgrup.

---

(1) Pojęcie izomorfizmu jest omówione szerzej na str. 187.

Streszczenie

Udowodniliśmy twierdzenie, że każda półgrupa z jednością jest izomorficzna z półgrupą pewnych przekształceń jakiegoś zbioru w siebie.

Zadania

1. Udowodnić, że każda półgrupa (nawet bez jedności) może być odwzorowana homomorficznie na półgrupę pewnych przekształceń jakiegoś zbioru w siebie tak, że działania zostają zachowane. Podać przykład, że przekształcenie to nie musi być izomorfizmem, gdyż nie musi być wzajemnie jednoznaczne.

2. Udowodnić twierdzenie Cayleya, że każda grupa jest izomorficzna z grupą pewnych przekształceń wzajemnie jednoznacznych jakiegoś zbioru na siebie.

3. Niech półgrupa P nie ma jedności. Dołączmy do zbioru P element E i określmy mnożenie na P tak jak poprzednio, zaś $E\cdot X = X = X\cdot E$, dla każdego X należącego do P, oraz $E\cdot E = E$. Udowodnić, że zbiór $P' = \{P, E\}$ z tak określonym mnożeniem tworzy półgrupę, której E jest jednością.

4. Opierając się na wyniku zadania 3, udowodnić, że każda półgrupa nawet bez jedności, jest izomorficzna z półgrupą pewnych przekształceń jakiegoś zbioru w siebie.

Wskazówka. Jest to uogólnienie twierdzenia 4. Przy dowodzie należy zamiast zbioru (1) wziąć zbiór elementów P'.

§ 46. IZOMORFIZMY, HOMOMORFIZMY

Rozpatrzmy teraz ważne i ciekawe pojęcie algebraiczne — pojęcie homomorfizmu. Weźmy dwie półgrupy P i Q oraz odwzorowanie φ przyporządkowujące każdemu elementowi $X\in P$ jakiś element $Y\in Q$. Będziemy mówili, że odwzorowanie jest *homomorfizmem* półgrupy P w półgrupę Q, jeżeli

$$\varphi(X\cdot X') = \varphi(X)\cdot\varphi(X') \text{ dla każdych } X,\ X'\in P. \tag{1}$$

Zwróćmy uwagę, że kropka po lewej stronie oznacza mnożenie elementów z P, kropka zaś po prawej stronie oznacza mnożenie elementów z Q.

Często mówimy krótko, choć niezbyt precyzyjnie, że homomorfizm jest to takie odwzorowanie, które iloczyn elementów przeprowadza na iloczyn elementów.

Na przykład odwzorowanie φ opisane w poprzednim paragrafie przyporządkowujące każdemu elementowi X półgrupy P element F_X półgrupy F jest homomorfizmem P w F.

Inny przykład homomorfizmu otrzymujemy, biorąc takie odwzorowanie φ półgrupy macierzy $\mathbf{M}_n$ w półgrupę liczb rzeczywistych, które każdej macierzy $\mathbf{A}\in\mathbf{M}_n$ przyporządkowuje wyznacznik $|\mathbf{A}|$ tej macierzy. A więc $\varphi(\mathbf{A}) = |\mathbf{A}|$. Zgodnie z twierdzeniem Cauchy'ego

$$|\mathbf{A}\cdot\mathbf{B}| = |\mathbf{A}|\cdot|\mathbf{B}|$$

(kropka po lewej stronie oznacza mnożenie macierzy, kropka zaś po prawej stronie mnożenie wyznaczników czyli mnożenie liczb), przekształcenie φ przeprowadza iloczyn macierzy na iloczyn odpowiadających im liczb

$$\varphi(\mathbf{A}\cdot\mathbf{B}) = \varphi(\mathbf{A})\cdot\varphi(\mathbf{B}),$$

a więc jest homomorfizmem.

Spośród homomorfizmów największe znaczenie mają homomorfizmy wzajemnie jednoznaczne jednej półgrupy na drugą. Homomorfizmy takie nazywają się *izomorfizmami*. O izomorfizmach mówiliśmy już w § 41.

Półgrupy, dla których można ustalić izomorfizm, nazywają się *półgrupami izomorficznymi*. Półgrupy takie są z algebraicznego punktu widzenia nieodróżnialne. Mają wszystkie własności takie same. Każda formuła zbudowana z symbolu „=", predykatu równoważności i symbolu działania „·" (składania) prawdziwa w jakiejś półgrupie jest prawdziwa w półgrupie z nią izomorficznej. Zajmijmy się głębiej tym zagadnieniem.

Rozpatrzmy wyrażenia postaci

$$f(x_1, x_2, \ldots, x_n) = g(x_1, x_2, \ldots, x_n), \tag{2}$$

gdzie $f(x_1, \ldots, x_n)$ i $g(x_1, \ldots, x_n)$ są termami zbudowanymi ze zmiennych $x_1, \ldots, x_n$ i znaku działania „·" (składania). Wyrażenia postaci (2) są najprostszymi formułami (formułami atomowymi) teorii półgrup. Z tych formuł budujemy inne łącząc je spójnikami zdaniowymi i opatrując kwantyfikatorami (zob. rozdział 4 paragraf 23).[1]

[1] Często rozpatrując teorię półgrup z jednością (lub zerem) względnie zerem i jednością oprócz termów zbudowanych ze znaku działania „·" (składania) dopuszczamy termy, w których występują działania zeroargumentowe, czyli stałe: E (jedność) względnie 0 (zero).

Po wstawieniu w formule (2) za zmienne elementów $A_1, A_2, \ldots$ jakiejś półgrupy P, otrzymujemy zdania prawdziwe w P bądź nie. Na przykład wstawiając do wyrażenia

$$(x_1 \cdot x_2) \cdot x_3 = x_1 \cdot (x_2 \cdot x_3)$$

elementy A, B, C półgrupy P, otrzymamy zdanie

$$(A \cdot B) \cdot C = A \cdot (B \cdot C)$$

prawdziwe, dzięki prawu łączności, niezależnie od tego jakie wzięliśmy elementy A, B, C z półgrupy P.

Rozpatrując teorię elementarną półgrup, spośród formuł mamy wyróżnione te zdania, czyli formuły bez zmiennych wolnych, które są prawdziwe w półgrupie P. Oznaczmy zbiór tych zdań przez $E(P)$.

Dla dwóch półgrup P i Q mamy w ten sposób dwa zbiory zdań $E(P)$ oraz $E(Q)$. Jeżeli półgrupy P i Q są izomorficzne to zbiory te pokrywają się.

Oczywiście zbiór $E(P)$ zawiera wszystkie twierdzenia teorii półgrup. Zawiera na ogół i inne zdania prawdziwe w półgrupie P, ale fałszywe w innych półgrupach, a więc na pewno nie będące twierdzeniami teorii półgrup. Na ogół nie można zbioru zdań $E(P)$ zaksjomatyzować, dodając do aksjomatów teorii półgrup pewną liczbę nowych aksjomatów tak, by zbiór $E(P)$ był zbiorem twierdzeń nowo uzyskanej teorii. Ponadto dwie półgrupy P i Q nie izomorficzne mogą mieć te same zbiory $E(P)$ i $E(Q)$. Mówi się też, że izomorfizmu półgrup nie można opisać w języku teorii elementarnej. Wyjątek stanowi przypadek, gdy półgrupy są skończone (por. zad. 9).

Streszczenie

Zdefiniowaliśmy pojęcie homomorfizmu i izomorfizmu półgrup i omówiliśmy jego własności.

Zadania

1. Wskazać półgrupę bez zera złożoną z macierzy i taki jej obraz homomorficzny, który jest półgrupą z zerem.

2. Udowodnić, że jeżeli P oraz Q są dwoma półgrupami, φ zaś homomorfizmem takim, że $\varphi(P) = Q$, to obraz $E' = \varphi(E)$ jedności E półgrupy P będzie jednością półgrupy Q.

3. Udowodnić, że jeżeli φ jest homomorfizmem półgrupy P w Q, ψ zaś homomorfizmem półgrupy Q w R, to złożenie $\psi\varphi$ jest homomorfizmem półgrupy P w R.

4. Niech $\Phi(P)$ oznacza zbiór homomorfizmów półgrupy P w siebie. Udowodnić, że zbiór $\Phi(P)$ tworzy półgrupę, w której mnożeniem jest składanie homomorfizmów.

5. Półgrupa P składa się z dwóch elementów, których mnożenie opisane jest za pomocą tabelki:

	A	B
A	A	A
B	A	B

Ile elementów ma półgrupa $\Phi(P)$ (zob. zadanie poprzednie). Opisać tę półgrupę, wskazując elementy i tabliczkę działania.

6. Niech φ będzie izomorfizmem półgrupy P z półgrupą Q. Określmy przekształcenie ψ zbioru Q na P takie, że $\psi(B) = A$, gdzie A takie, że $\varphi(A) = B$.

Udowodnić że:

a) ψ jest przekształceniem wzajemnie jednoznacznym odwzorowującym Q na P.

b) ψ jest homomorfizmem.

Wynika stąd, że ψ jest izomorfizmem półgrupy Q na P.

7. Udowodnić, że relacja $P \sim Q$ zachodząca między półgrupami P i Q wtedy i tylko wtedy, gdy istnieje izomorfizm ψ przekształcający P na Q, jest równoważnością, tzn. jest zwrotna, symetryczna i przechodnia.

8. Izomorfizm półgrupy na siebie nazywa się *automorfizmem*. Udowodnić, że zbiór $A(P)$ wszystkich automorfizmów półgrupy P na siebie tworzy grupę ze składaniem przekształceń jako działaniem grupowym. Grupa ta nazywa się *grupą automorfizmów* półgrupy P.

9. Udowodnić, że dla półgrupy P skończonej istnieje taki skończony zbiór aksjomatów A teorii $E(P)$, że każda półgrupa będąca modelem teorii $E(P)$ będzie izomorficzna z P. Wskazówka. Jeżeli $A_1, A_2, \ldots, A_n$ są wszystkimi elementami podgrupy, to działanie opisuje się za pomocą tabelki. Tabelka taka jest właściwie innym zapisem n równości $A_1 \cdot A_j = A_k$. Zbiór A będzie składał się z aksjomatów teorii półgrup uzupełnionej jednym aksjomatem:

$$\begin{aligned} &Ex_1 Ex_2 \ldots Ex_n\{Ay[(y = x_1) \vee (y = x_2) \vee \ldots \vee (y = x_n)] \,\& \\ &\quad \& (x_1 \neq x_2) \,\& (x_1 \neq x_3) \,\& \ldots \,\& (x_1 \neq x_n) \,\& (x_2 \neq x_3) \,\& \ldots \,\& \\ &\quad \& (x_2 \neq x_n) \,\& \ldots \,\& (x_{n-1} \neq x_n) \,\& (x_1 \cdot x_1 = x_t) \,\& \ldots \,\& (x_n \cdot x_n = x_c)\}. \end{aligned}$$

§ 47. KONGRUENCJE. PÓŁGRUPY ILORAZOWE

Paragraf ten jak i dwa następne są nieco trudniejsze. Jednak z uwagi na wagę paragrafu 49 traktującego o tzw. zagadnieniu słów dla spraw maszynowych, zaleca się cierpliwe i staranne zapoznanie się z ich treścią.

Mówiąc o aksjomatyce teorii półgrup, mówiliśmy o relacji równości „=", od której wymagaliśmy by była zwrotna, symetryczna i przechodnia oraz by związana była z mnożeniem za pomocą następującego prawa:

$$\text{jeżeli } A = A' \text{ i } B = B', \text{ to } A \cdot B = A' \cdot B'.$$

Każdą relację „$\sim$" określoną w półgrupie P

zwrotną: $A \sim A$,

symetryczną: jeżeli $A \sim B$, to $B \sim A$,

przechodnią: jeżeli $A \sim B$ i $B \sim C$, to $A \sim C$,

i taką, że

(1) $$\text{jeżeli } A \sim A' \text{ i } B \sim B', \text{ to } A \cdot B \sim A' \cdot B',$$

nazywamy *kongruencją*. Innymi słowy, relację równoważności nazywamy kongruencją, jeżeli mnożenie równoważnych elementów daje równoważne wyniki.

Na przykład relacja „$\sim$" określona w półgrupie N liczb naturalnych z działaniem mnożenia

$$n \sim m \text{ wtedy i tylko wtedy, gdy } n = m \pmod{s},$$

gdzie s jest ustaloną liczbą naturalną, jest kongruencją.

Podamy jeszcze jeden bardzo ważny przykład. Treść jego zawarta jest w twierdzeniu następującym

TWIERDZENIE 5. *Jeżeli P jest półgrupą, φ zaś jej homomorfizmem na półgrupę Q, to relacja „$\underset{\varphi}{\sim}$" określona następująco: $A \underset{\varphi}{\sim} B$ wtedy i tylko wtedy, gdy $\varphi(A) = \varphi(B)$ w półgrupie Q, jest kongruencją w półgrupie P.*

Z definicji wynika, że relacja „$\underset{\varphi}{\sim}$" jest równoważnością. Dla dowodu (1) załóżmy, że $A \underset{\varphi}{\sim} A'$ i $B \underset{\varphi}{\sim} B'$. Wtedy $\varphi(A) = \varphi(A')$ i $\varphi(B) = \varphi(B')$, a więc ponieważ φ jest homomorfizmem więc i $\varphi(A \cdot B) = \varphi(A' \cdot B')$. Wynika stąd żądana teza

$$A \cdot B \underset{\varphi}{\sim} A' \cdot B'.$$

Rozważmy kongruencję „$\sim$" jakiejś półgrupy P. Kongruencję tą można przyjąć za relację równości w klasach abstrakcji relacji. Otrzymamy wtedy z półgrupy P nową półgrupę, w której wszystkie elementy kongruentne zostaną utożsamione z jednym elementem — klasą abstrakcji relacji „$\sim$" (por. rozdział 6, § 34). Dzięki warunkowi (1) definiującemu kongruencję, wyniki mnożenia wykonanego na równych — a więc poprzednio kongruentnych elementach będą równe.

Tę nową półgrupę, której elementami są klasy abstrakcji relacji „$\sim$" półgrupy P, nazywamy *półgrupą ilorazową* półgrupy P względem relacji kongruencji „$\sim$" i oznaczamy symbolem $P/\sim$. Opiszmy tę półgrupę nieco dokładniej.

Niech A, B, C, będą elementami półgrupy P. Elementami półgrupy $P/\sim$ będą klasy abstrakcji $[A]$, $[B]$, $[C]$ relacji „$\sim$", a działanie mnożenia w $P/\sim$, czyli mnożenia klas abstrakcji określone jest następująco

(2) $$[A] \cdot [B] = [A \cdot B].$$

Czytelnikowi zostawiamy w oparciu o definicję klas abstrakcji i własność (1) relacji „$\sim$", dowód, że działanie „$\cdot$" mnożenia klas abstrakcji, wykonane na równych elementach daje równe wyniki, oraz dowód tego, że działanie to jest łączne (por. zadanie 2).

Udowodnimy teraz następujące twierdzenie

TWIERDZENIE 6. *Odwzorowanie $\varphi/\sim$, przyporządkowujące każdemu X z półgrupy P element $\varphi_\sim(X) = [X]$ z półgrupy $P/\sim$, jest homomorfizmem P na $P/\sim$.*

Homomorfizm ten nazywamy *naturalnym homomorfizmem P na $P/\sim$*, a półgrupę $P/\sim$ półgrupą ilorazową.

Dowód. Oczywiście z definicji $P/\sim$ wynika, że $\varphi_\sim$ jest odwzorowaniem na całą półgrupę $P/\sim$. Odwzorowanie to spełnia warunek (1) z § 46

$$\varphi_\sim(X) \cdot \varphi_\sim(X') = \varphi_\sim(X \cdot X'),$$

gdyż $[X] \cdot [X'] = [X \cdot X']$, zgodnie z definicją 2 mnożenia klas abstrakcji

Na zakończenie tego paragrafu podamy jeszcze następujące twierdzenie, zwane twierdzeniem o homomorfizmie.

TWIERDZENIE 7. *Jeżeli φ jest homomorfizmem półgrupy P na półgrupę Q φ zaś kongruencją określoną tak jak w twierdzeniu 5, to przyporządkowani*

ψ każdej klasie abstrakcji relacji „$\underset{\varphi}{\sim}$” (tzn. każdemu elementowi półgrupy $P/\underset{\varphi}{\sim}$), elementu półgrupy Q tak, że

$$\psi([X]) = \varphi(X)$$

jest izomorfizmem. Na odwrót, jeżeli $\varphi_\sim$ jest naturalnym homomorfizmem wyznaczonym tak jak w twierdzeniu 6 przez kongruencję „$\sim$”, to kongruencja „$\underset{\varphi_\sim}{\sim}$” wyznaczona przez ten homomorfizm jest identyczna z kongruencją „$\sim$”.

Twierdzenia tego nie będziemy tutaj dowodzić, gdyż nie chodzi nam o subtelności techniczne dowodu. Twierdzenie 5 mówi, że każdy homomorfizm wyznacza pewną kongruencję, twierdzenie 6 zaś mówi, że każda kongruencja wyznacza pewien homomorfizm. Twierdzenie 7 stwierdza, że odpowiedniość ta jest wzajemnie jednoznaczna. Dzięki temu twierdzeniu możemy badanie homomorfizmów zastąpić badaniem kongruencji.

Streszczenie

Podaliśmy definicję kongruencji i półgrupy ilorazowej.

Zadania

1. Udowodnić, że relacja równoważności „$\sim$” w półgrupie P taka, że: $A \sim A'$ pociąga $AB \sim A'B$ i $BA \sim B'A$ dla każdego $A, A', B, B' \in P$ jest kongruencją.

2. a) Udowodnić, że jeżeli $[A] = [A']$ i $[B] = [B']$, to $[A]\cdot[B] = [A']\cdot[B']$.

Wskazówka. Z definicji klas abstrakcji wynika, że $[A] = [A']$ wtedy i tylko wtedy, gdy $A \sim A'$ (por. rozdz. V par. 28).

b) Udowodnić, że:

$$[A]\cdot([B]\cdot[C]) = ([A]\cdot[B])\cdot[C] = [(A\cdot B)\cdot C] = [A\cdot(B\cdot C)].$$

3. Jeżeli „$\sim$” jest kongruencją w półgrupie N liczb naturalnych z mnożeniem określoną następująco: $n \sim m$ wtedy i tylko wtedy, gdy $n = m(\mathrm{mod}\, s)$, to półgrupę N nazywamy *półgrupą reszt modulo s*.

a) Udowodnić, że $N/\sim$ ma dokładnie s elementów. Opisać te elementy.

b) Ułożyć tabelkę mnożenia w półgrupie $N/\sim$ dla $s = 3$.

4. Niech $\Xi(P)$ będzie zbiorem wszystkich kongruencji półgrupy P. Określmy relację „$\leqslant$” w zbiorze $\Xi(P)$ następująco:

$$\sim_1 \leqslant \sim_2$$

wtedy i tylko wtedy, gdy dla każdych X i Y z tego, że $X \sim_1 Y$ wynika $X \sim_2 Y$.

a) Udowodnić, że relacja $a/\leqslant$ jest relacją porządku w zbiorze $\Xi(P)$.

b) Zbiór $\Xi(P)$ tworzy siatkę (strukturę) względem relacji „$\leqslant$”.

c) Siatka ta ma element minimalny: relację równości i element maximalny, kongruencję, która utożsamia wszystkie elementy półgrupy P (por. zad. 6 rozdział V, § 30).

5. Półgrupa P składa się z elementów A, B, C, D i mnożenie jest opisane za pomocą tabelki

	A	B	C	D
A	A	A	A	A
B	A	B	C	D
C	A	C	C	A
D	A	D	A	D

a) Wskazać jedność E i zero 0 tej półgrupy. Udowodnić, że dla kongruencji „$\sim$” jeżeli $C \sim D$, to $C \sim 0$ i $D \sim 0$. Jeżeli $E \sim 0$ lub $C \sim E$, lub $D \sim E$, to dla każdego X z tabelki $X \sim 0$.

b) Opisać wszystkie kongruencje półgrupy P.

c) Narysować wykres siatki $\Xi(P)$.

6. Udowodnić twierdzenie 7.

7. a) Udowodnić, że jeżeli dla dwóch kongruencji półgrupy P $\sim_1 \leqslant \sim_2$ (por. zad. 4), to istnieje homomorfizm ψ półgrupy $P/\sim_1$ na półgrupę $P/\sim_2$ taki, że

$$\psi\cdot\varphi_{\sim_1} = \varphi_{\sim_2},$$

8. Jeżeli P jest półgrupą zaś „$\approx$” dowolną relacją określoną dla półgrupy P, to istnieje kongruencja „$\sim$” taka, że:

(*) jeżeli $A \approx B$, to $A \sim B$.

Kongruencją taką jest na przykład kongruencja utożsamiająca wszystkie elementy półgrupy P. Udowodnić, że w zbiorze $R \subset \Xi(P)$ (por. zad. 4) istnieje kongruencja najmniejsza o własności (*). Kongruencję tę nazywamy *rozszerzeniem relacji* „$\approx$”, do kongruencji.

§ 48. PÓŁGRUPY WOLNE

Omówimy teraz bardzo ważny, zarówno ze względu na zastosowanie praktyczne jak i rozważania teoretyczne przykład półgrupy.

Weźmy sobie jakikolwiek zbiór niepusty elementów, oznaczmy go przez X

$$X = \{X_1, X_2, X_3, \ldots\}$$

i przez $\Pi(X)$ oznaczmy zbiór wszystkich ciągów dowolnej skończonej długości:

$$U = Y_1 Y_2 \ldots Y_n,$$

których wyrazy $Y_1, Y_2, \ldots$ są elementami ze zbioru X.

Ciągi zapisujemy tutaj pisząc wyrazy kolejno bez oddzielania przecinkami. Liczbę n nazywamy *długością ciągu* U. Ciągi długości 1 składają się z pojedynczych symboli.

Określmy w zbiorze $\Pi(X)$ relację równoważności następująco: Dwa ciągi różnej długości będą zawsze nierównoważne, ciągi zaś U i V równej długości

$$U = Y_1 Y_2 \ldots Y_n \quad \text{i} \quad V = Z_1 Z_2 \ldots Z_n$$

będą równoważne wtedy i tylko wtedy, gdy:

$$Y_1 = Z_1, \quad Y_2 = Z_2, \quad \ldots, \quad Y_n = Z_n.$$

(Relacja ta jest więc po prostu relacją równości ciągów).

Działanie składania ciągów określimy następująco:

Dla dwóch ciągów $U = Y_1 Y_2 \ldots Y_n$ i $V = Z_1 Z_2 \ldots Z_m$ iloczynem $U \cdot V$ będziemy nazywać ciąg długości $n+m$:

$$W = Y_1 Y_2 \ldots Y_n Z_1 Z_2 \ldots Z_m.$$

Definicję składania ciągów można by napisać następująco:

$$(Y_1\ Y_2 \ldots Y_n)\cdot(Z_1\ Z_2 \ldots Z_m) = Y_1\ Y_2 \ldots Y_n\ Z_1\ Z_2 \ldots Z_m$$

Działanie to nosi często nazwę *konkatenacji (zetknięcia)* dwóch ciągów.

Dowód łączności działania konkatenacji ciągów nie przedstawia żadnych trudności.

Zbiór $\Pi(X)$ jest więc przy takim określeniu działań półgrupą. Elementy $X_1, X_2, X_3, \ldots$ zbioru X, są jak już zauważyliśmy ciągami o długości 1, a więc należą do $\Pi(X)$. Co więcej każdy element $\Pi(X)$ można otrzymać z tych elementów, jako wynik wykonania skończonej ilości składań.

Półgrupę $\Pi(X)$ nazywamy *półgrupą wolną*, zbiór X zaś jej *zbiorem wolnych tworzących*. Elementy półgrupy $\Pi(X)$ nazywamy *słowami* $X_1, X_2, X_3, \ldots$

Zachodzi następujące bardzo ważne twierdzenie.

TWIERDZENIE 8. *Dla dowolnej półgrupy P i dowolnego odwzorowania μ zbioru X w zbiór elementów półgrupy P, istnieje homomorfizm φ półgrupy $\Pi(X)$ w P taki, że $\varphi(Y) = \mu(Y)$ dla $Y \in X$.*

Treść tego twierdzenia wyrażamy krótko, mówiąc, że każde odwzorowanie zbioru wolnych tworzących w dowolną półgrupę można rozszerzyć do homomorfizmu całej półgrupy wolnej,

Dowód. Niech odwzorowanie μ będzie takie, że:

$$\mu(X_1) = A_1, \quad \mu(X_2) = A_2, \ldots$$

Określmy odwzorowanie φ następująco: dla $U \in \Pi(X)$

$$U = X_{a_1} X_{a_2} \ldots X_{a_n} \quad \text{jest } \varphi(U) = A_{a_1} A_{a_2} \ldots A_{a_n}.$$

Jasne jest, że $\varphi(X_1) = A_1 = \mu(X_1)$, $\varphi(X_2) = A_2 = \mu(X_2)$, ... Z definicji konkatenacji mamy

$$\varphi(U \cdot V) = \varphi(U) \cdot \varphi(V),$$

co kończy dowód.

Jako wniosek z twierdzenia 8 podamy następujące twierdzenie:

TWIERDZENIE 9. *Każda półgrupa P jest homomorficznym obrazem pewnej półgrupy wolnej $\Pi(X)$* (por. zadanie 4).

W innym sformułowaniu można by to twierdzenie wypowiedzieć następująco. Dla każdej półgrupy P istnieje taka relacja kongruencji „$\sim$" określona na pewnej półgrupie wolnej $\Pi(X)$, że półgrupa P jest izomorficzna z półgrupą ilorazową $\Pi(X)/\sim$ (por. tw. 5 i 7).

Z twierdzenia tego wynika, że każdą półgrupę P można określić przez podanie zbioru X wolnych generatorów półgrupy $\Pi(X)$, oraz kongruencji „$\sim$" takiej, że $\Pi(X)/\sim$ jest izomorficzne z P.

Problem tkwi w tym jak określić kongruencję „$\sim$". Najprościej to zrobić przez wypisanie wszystkich par elementów półgrupy $\Pi(X)$ kongruentnych ze sobą. Trudność polega na tym, że półgrupa ta ma nieskończenie wiele elementów (por. zadanie 1) musielibyśmy więc, chcąc opisać kongruencję „$\sim$", podać nieskończenie wiele par (U, V), gdzie U, V są ciągami symboli z X (elementami półgrupy $\Pi(X)$) takich, że

$$U \sim V.$$

Musielibyśmy mianowicie podać wszystkie takie pary.

Zbadamy teraz jak można tu uniknąć podawania wszystkich takich par a zadowolić się podawaniem tylko niektórych. Na wstępie podamy kilka definicji.

DEFINICJA. Zbiór R wszystkich par (U, V), gdzie $U, V \in \Pi(X)$ takich, że $(U, V) \in R$ wtedy i tylko wtedy, gdy $U \sim V$, nazywamy *pełnym układem relacji* określających półgrupy $\Pi(X)/\sim$.

Łatwo można udowodnić następujące

TWIERDZENIE 10. *Jeżeli R jest pełnym układem relacji określających półgrupy $\Pi(X)/\sim$, to:*

1. $(U, U) \in R$ *dla każdego* $U \in \Pi(X)$;
2. *jeżeli* $(U, V) \in R$, *to* $(V, U) \in R$;
3. *jeżeli* $(U, V) \in R$ *i* $(V, W) \in R$, *to* $(U, W) \in R$;
4. *jeżeli* $(U, V) \in R$, *to dla każdych* $K, L \in \Pi(X)$ jest $(K \cdot U, K \cdot W) \in R$ *i* $(U \cdot L, V \cdot L) \in R$;
5. *jeżeli półgrupa* $\Pi(X)/\sim$ *jest półgrupą ze skracaniem lewostronnym, to jeżeli* $(K \cdot U, K \cdot V) \in R$, *to* $(U, V) \in R$;
6. *jeżeli półgrupa* $\Pi(X)/\sim$ *jest półgrupą ze skracaniem prawostronnym, to jeżeli* $(U \cdot L, V \cdot L) \in R$, *to* $(U \cdot V) \in R$.

Dowód 1-4 wynika z definicji relacji kongruencji (por. par. 46 tego rozdziału), dowód zaś 5 i 6 z definicji półgrupy ze skracaniem.

Prawdziwe jest również twierdzenie odwrotne.

TWIERDZENIE 11. *Jeżeli R jest zbiorem par (U, V), $U, V \in \Pi(X)$ spełniającym warunki* 1-4 *twierdzenie* 10, *to relacja $\underset{R}{\sim}$, określona następująco: $U \underset{R}{\sim} V$ wtedy i tylko wtedy, gdy $(U, V) \in R$, jest relacją kongruencji. Układ jest wtedy pełnym układem relacji określających półgrupy $\Pi(X)/\underset{R}{\sim}$*

Jeżeli zbiór R spełnia dodatkowo warunek 5 (względnie 6) to półgrupa $\Pi(X)/\sim$ jest półgrupą ze skracaniem lewostronnym (względnie prawostronnym). Literatura Malcew, Markow, Davis.

Streszczenie

Podaliśmy definicję półgrupy wolnej, elementów tworzących i relacji określających.

Zadania

1. Udowodnić, że jeżeli X jest zbiorem skończonym, to elementy półgrupy $\Pi(X)$ można ustawić w ciąg nieskończony.

2. Pokazać, że półgrupa $\Pi(X)$ nie ma jedności. Jedność tę można dołączyć do półgrupy, dodając do $\Pi(X)$ symbol Λ (tzn. ciąg pusty o długości 0) i określając

$$U \cdot \Lambda = \Lambda \cdot U = U \quad \text{dla} \quad U \in \Pi(X), \quad \Lambda \cdot \Lambda = \Lambda.$$

Tak rozszerzona półgrupa $\Pi^*(X)$ będzie półgrupą z jednością (por. zadanie 3 z par. 39 tego rozdziału).

3. Udowodnić, że jeżeli P jest półgrupą z jednością E, to odwzorowanie φ^* półgrupy $\Pi^*(P)$ określone w następujący sposób:

$$\varphi^*(U) = \varphi(U) \quad \text{dla} \quad U \neq \Lambda,$$

zaś

$$\varphi^*(\Lambda) = E$$

(gdzie φ jest określone tak jak w twierdzeniu 8) jest homomorfizmem półgrupy $\Pi^*(X)$ w P.

Wyprowadzić uogólnienie twierdzenia 8 dla półgrupy $\Pi^*(X)$.

4. Weźmy półgrupę P i zbiór X taki, że istnieje odwzorowanie wzajemnie jednoznaczne X na P. Udowodnić, że rozszerzenie odwzorowania φ do homomorfizmu półgrupy $\Pi(X)$, jest homomorfizmem $\Pi(X)$ na P.

5. Udowodnić twierdzenia 10 i 11, biorąc zamiast $\Pi(X)$ dowolną półgrupę ze skracaniem.

6. Udowodnić, że: jeżeli do warunków 1-4 twierdzenia 10 nałożonych na pełny zbiór relacji określających kongruencję

a) dołączymy warunek:

istnieje $W \in \Pi(X)$ takie, że dla dowolnego $K \in \Pi(X)$ $(K \cdot W, K) \in R$ i $(W \cdot K, K) \in R$, to półgrupa $\Pi(X)/\sim$ jest półgrupą z jednością:

b) dołączymy warunek:

istnieje $T \in \Pi(X)$, takie, że dla każdego $L \in \Pi(X)$ $(L \cdot T, T) \in R$ i $(T \cdot L, T) \in R$, o półgrupa $\Pi(X)/\sim$ jest półgrupą z zerem.

§ 49. ZAGADNIENIE SŁÓW

Rozpatrzmy teraz następujący przykład. Niech P będzie półgrupą złożoną z elementów A, B, E w której działanie mnożenia określone jest tabelką

	E	A	B
E	E	A	B
A	A	B	E
B	B	E	A

Tabelkę tę można opisać równaniami:

$$(1)\qquad A\cdot A=B,\quad A\cdot B=B\cdot A=E,\quad A\cdot E=E\cdot A=A,$$
$$B\cdot E=E\cdot B=B,\quad E\cdot E=E.$$

Z równości tych wynika, że E jest jednością półgrupy P, A zaś elementem tworzącym, to znaczy takim elementem, że każdy inny można otrzymać przez mnożenie: elementu A przez siebie, wyniku $A\cdot A$ przez A, i tak dalej. Zamiast równości (1) można by napisać prostszy układ równości

$$(2)\qquad A\cdot A\cdot A\cdot A=A,\quad A\cdot A=B,\quad A\cdot A\cdot A=E.$$

Weźmy teraz półgrupę wolną $\Pi(X)$, gdzie X jest zbiorem złożonym z jednego elementu Y. Elementami półgrupy $\Pi(X)$ będą wszystkie ciągi Y, YY, YYY, ...

Odwzorowanie μ takie, że $\mu(Y)=A$, daje się rozszerzyć do homomorfizmu φ półgrupy $\Pi(X)$ na P. Na przykład będzie $\varphi(Y)=A$, $\varphi(YY)=A\cdot A=B$, $\varphi(YYY)=A\cdot A\cdot A=E$, zaś $\varphi(YYYY)=A\cdot A\cdot A\cdot A=A$ i tak dalej.

Kongruencja $\underset{\varphi}{\sim}$ taka, że półgrupa $\Pi(X)/\underset{\varphi}{\sim}$ jest izomorficzna z P i będzie wyznaczona tylko przez jedną równość określającą

$$YYYY\sim Y,$$

którą w zapisie używanym w twierdzeniach 10 i 11 przedstawimy następująco:

para $(YYYY, Y)$ jest relacją określającą.

Pełny układ relacji określających dla tej kongruencji będzie zawierał parę $(YYYY, Y)$ i będzie najmniejszym układem spełniającym warunki 1-4 twierdzenia 10, zawierającym tę parę. Łatwo opisać ten układ.

Para (U, V) należy do R dla $U=YY\ldots Y$ długości n, i $V=YY\ldots YYY$ długości m wtedy i tylko wtedy, gdy liczby n i m różnią się od siebie wielokrotnością trójki.

Pełny układ relacji określających opisaliśmy w tym przypadku jedną prostą równością $YYYY\sim Y$.

Poprzednia uwaga pozwala orzec w każdym przypadku czy słowa w Y będą równe w półgrupie P. Mówiąc precyzyjniej mamy metodę roz-

strzygnięcia czy zdanie

„dla każdego y (jeżeli $y\cdot y\cdot y\cdot y=y$, to $y\cdot y\ldots y=y\cdot y\ldots y\cdot y\cdot y$)"

jest zdaniem teorii półgrup czy też nie, dla wyrażeń $yy\ldots y$ i $yy\ldots yyy$ dowolnej długości.

Postawmy problem ogólniej. Rozpatrzmy zbiór równości

$$(3)\qquad \begin{aligned} &f_1(x_1,\ldots,x_n)=g_1(x_1,\ldots,x_n),\\ &\ldots\ldots\ldots\ldots\ldots\ldots\\ &f_k(x_1,\ldots,x_n)=g_k(x_1,\ldots,x_n)\end{aligned}$$

i równość

$$(4)\qquad f(x_1,\ldots,x_n)=g(x_1,\ldots,x_n),$$

gdzie $f, g, f_1, \ldots, f_k, g_1, \ldots, g_k$ są wyrażeniami zbudowanymi ze zmiennych $x_1, \ldots, x_n$ i znaku składania. Zagadnienie, czy zdanie

(5) dla każdych $x_1, \ldots, x_n$ {jeżeli $[f_1(x_1,\ldots,x_n)=g_1(x_1,\ldots,x_n), \ldots, f_k(x_1,\ldots,x_n)=g_k(x_1,\ldots,x_n)]$, to $f(x_1,\ldots,x_n)=g(x_1,\ldots,x_n)$};

jest twierdzeniem teorii czy nie, nazywa się *zagadnieniem słów*. Jest to zagadnienie, czy z równości (3) wynika równość (4).

Udowodnimy następujące twierdzenie

TWIERDZENIE 12. *Wyrażenie* (5) *jest twierdzeniem teorii półgrup wtedy i tylko wtedy, gdy równość*

$$f(x_1,\ldots,x_n)=g(x_1,\ldots,x_n)$$

można otrzymać z równości (3) *i równości typu*

$$(6)\qquad h(x_1,\ldots,x_n)=h(x_1,\ldots,x_n),$$

gdzie h dowolne wyrażenie zbudowane ze zmiennych $x_1, \ldots, x_n$ i znaku mnożenia przez skończoną liczbę operacji (7), (8), (9):

(7) *przejście od równości* $k(x_1,\ldots,x_n)=l(x_1,\ldots,x_n)$ *do równości* $l(x_1,\ldots,x_n)=k(x_1,\ldots,x_n)$;

(8) *przejście od równości* $k(x_1,\ldots,x_n)=l(x_1,\ldots,x_n)$ *i* $l(x_1,\ldots,x_n)=t(x_1,\ldots,x_n)$ *do równości* $k(x_1,\ldots,x_n)=t(x_1,\ldots,x_n)$;

(9) *przejście od równości* $k(x_1, \ldots, x_n) = l(x_1, \ldots, x_n)$,

do równości $y \cdot k(x_1, \ldots, x_n) = y \cdot l(x_1, \ldots, x_n)$

lub równości $k(x_1, \ldots, x_n) \cdot y = l(x_1, \ldots, x_n) \cdot y$,

gdzie y *jest którąkolwiek ze zmiennych* $x_1, \ldots, x_n$.

Dowód. Oznaczmy przez E najmniejszy zbiór równości, zawierający równości (3) i wszystkie równości typu (6) i zamknięty ze względu na operacje (7), (8), (9).

Jeżeli równość $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ należy do E to zdanie (5) jest twierdzeniem teorii, gdyż jest konsekwencją aksjomatów równości i teorii półgrup.

Rozważmy przypadek, gdy równość

(10) $$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$$

nie należy do E.

Weźmy półgrupę wolną $\Pi(X)$, gdzie X jest zbiorem złożonym z elementów $X_1, \ldots, X_n$. Mamy następujący układ relacji określających:

para $\langle k(X_1, \ldots, X_n), l(X_1, \ldots, X_n)\rangle$ należy do R wtedy i tylko wtedy, gdy równość $k(x_1, \ldots, x_n) = l(x_1, \ldots, x_n)$ należy do E (spełnia jak łatwo widać warunki 1-4 twierdzenia 10). Na mocy twierdzenia 11, E jest pełnym układem określającym półgrupy $\Pi(X)/\widetilde{R}$.

Jeżeli (10) nie należy do E to, z uwagi na to co powiedziano w poprzednim akapicie, kongruencja $f(X_1, \ldots, X_n) \widetilde{R} g(X_1, \ldots, X_n)$ nie zachodzi. Elementy $f(X_1, \ldots, X_n)$ i $g(X_1, \ldots, X_n)$ (a dokładniej ich klasy abstrakcji dla relacji „$\widetilde{R}$"), są w półgrupie $\Pi(X)/\widetilde{R}$ różne, choć wobec zachodzenia kongruencji

$$f_1(X_1, \ldots, X_n) \widetilde{R} g_1(X_1, \ldots, X_n), \quad \ldots, \quad f_k(X_1, \ldots, X_n) \widetilde{R} g_k(X_1, \ldots, X_n)$$

w półgrupie tej zachodzą równości

$$f_1(X_1, \ldots, X_n) = g_1(X_1, \ldots, X_n), \quad \ldots, \quad f_k(X_1, \ldots, X_n) = g_k(X_1, \ldots, X_n).$$

Zdanie (5) jest więc fałszywe w modelu będącym półgrupą $\Pi(X)/\widetilde{R}$ nie jest więc twierdzeniem teorii półgrup.

Dowód twierdzenia 12 został więc zakończony.

Twierdzenie 12 umożliwia więc badanie, czy z równości słów (3) wynika równość słów (4). Jeżeli tak jest (tzn. zdanie (5) jest twierdzeniem teorii), to twierdzenie 12 mówi, że wykonując kolejno różne możliwe operacje (7), (8), (9) na słowach (3) otrzymamy słowo (4). Jeżeli jednak wykonując nawet bardzo wiele takich operacji słowa (4) nie otrzymamy, to nie będziemy pewni, czy wykonując jeszcze pewną liczbę takich operacji nie otrzymamy słowa (4). Inaczej mówiąc nie będziemy mogli stwierdzić czy zdanie (5) nie jest zdaniem teorii.

Zachodzi następujące ważne twierdzenie, które zacytujemy bez dowodu. (Lit. Malcew, Markow, Davis).

TWIERDZENIE 13. *Zbiór zdań typu* (5) *jest nierozstrzygalny. Nie ma algorytmu, który pozwalał by stwierdzić, czy dowolne zdanie typu* (5) *jest twierdzeniem teorii czy też nie.*

Twierdzenie to ma pewne ważne praktyczne konsekwencje. Jeżeli mamy jakąś półgrupę P, mającą elementy $A_1, \ldots, A_n$ i być może jeszcze jakieś inne i jeżeli mamy w półgrupie tej równości określające

$$f_1(A_1, \ldots, A_n) = g_1(A_1, \ldots, A_n), \quad \ldots, \quad f_k(A_1, \ldots, A_n) = g_k(A_1, \ldots, A_n),$$

to na ogół nie mamy żadnej ogólnej metody pozwalającej przy dowolnych $t_1, \ldots, f_k, g_1, \ldots, g_k, f, g$ stwierdzić czy elementy

$$f(A_1, \ldots, A_n) \text{ i } g(A_1, \ldots, A_n)$$

są równe, czy też nie.

W przykładzie podanym na początku tego paragrafu była metoda sprawdzania czy dowolne dwa elementy półgrupy, zapisane za pomocą iloczynu elementów A, są równe, czy też nie.

Ale na przykład dla następującej półgrupy mającej elementy tworzące A, B, C, D, F i równości określające:

$$AC = CA, \quad BD = DB,$$
$$AD = DA, \quad FCA = AF,$$
$$BC = CB, \quad FDB = BF,$$
$$ABAC = ABACF$$

nie ma metody stwierdzenia, dla dowolnych dwóch słów f i g czy elementy $f(A, B, C, D, F)$ i $g(A, B, C, D, F)$ są równe czy też nie (zob. Ceitin 3).

Streszczenie

Zajmowaliśmy się wyznaczeniem półgrupy przez podanie elementów tworzących i relacji określających i zagadnieniem równości słów. Zagadnienie równości słów jest nierozstrzygalne.

Zadania

1. Udowodnić, posługując się aksjomatami równości i aksjomatami teorii półgrup, że jeżeli $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ należy do E, to zdanie (5) można wyprowadzić za pomocą aksjomatów.

2. Udowodnić, że jeżeli zbiór E^* jest najmniejszym zbiorem zamkniętym na operacje (6)—(9) i na operację przejścia od równości

$$h(x_1, \ldots, x_n)\, k(x_1, \ldots, x_n)\, m(x_1, \ldots, x_n) = h(x_1, \ldots, x_n)\, l(x_1, \ldots, x_n)\, m(x_1, \ldots, x_n)$$

do równości

$$k(x_1, \ldots, x_n) = l(x_1, \ldots, x_n),$$

to równość $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ należy do E wtedy i tylko wtedy, gdy zdanie (5) jest twierdzeniem teorii półgrup ze skracaniem.

Uwaga. Zagadnienie słów dla teorii półgrup ze skracaniem jest nierozstrzygalne. Zob. Malcew, Markow, Davis.

3. Niech P będzie półgrupą przemienną z jednością E o elementach tworzących $A_1, \ldots, A_n$ i relacjach określających

$$A_1^{m_1} = E, \quad \ldots, \quad A_n^{m_n} = E.$$

Udowodnić, że P ma dokładnie $m_1 \cdot m_2, \ldots, m_n$ elementów. Podać warunek konieczny i dostateczny na to, żeby dwa słowa były równe.

Rozdział 9

ALGORYTMY

Wiele działów matematyki znajduje ostatnio różne zastosowania pozamatematyczne, głównie w teorii maszyn matematycznych. Wydaje się jednakże, iż dzięki maszynom matematycznym największe zainteresowanie budzi pojęcie algorytmu. Pojęcie to przeżywa swą drugą — a właściwie trzecią — młodość.

Pierwszy etap rozwoju pojęcia algorytmu liczy już kilka tysięcy lat. Rozpoczął się on chyba wraz z powstaniem pierwszych pojęć matematycznych i trwał mniej więcej do roku 1930. Matematyka tego okresu polegała głównie na poszukiwaniu algorytmów rozwiązywania różnych zadań. W starożytności, wraz z rozwojem pojęcia liczby, rozwijały się metody wykonywania działań na liczbach, dodawania, odejmowania, a później mnożenia i dzielenia. Dalszy okres to algorytmy rozwiązywania równań algebraicznych, równań różniczkowych, czy też układów równań liniowych. Równolegle również poszukiwano algorytmów rozwiązywania różnych problemów geometrycznych. Przykładem jest znany powszechnie problem trysekcji kąta. We wszystkich tych przypadkach chodziło o podanie metody rozwiązania określonej klasy problemów. Wydawało się wówczas bowiem, że znalezienie odpowiedniej metody do rozwiązywania jakiegoś zagadnienia jest tylko kwestią czasu. Jeżeli nie jest ona znana dziś, sądzono, że na pewno znajdzie się ją później.

Drugi okres, należałoby właściwie zacząć liczyć od Hilberta, który postawił ogólne problemy dotyczące istnienia algorytmów rozwiązywania pewnych problemów — sam zresztą prawdopodobnie nie podejrzewając, że mogą one nie istnieć. Za właściwy początek tego okresu należy jednak

uważać intensywny rozwój podstaw matematyki w latach trzydziestych naszego stulecia, a w szczególności rezultaty Gödla, Turinga, Posta i inne dotyczące niezupełności i nierozstrzygalności sformalizowanych teorii matematycznych. Pojęcie algorytmu w tych badaniach odgrywa centralną rolę. Po burzliwym początku nastąpił jednak dość szybko okres wyczerpania się tej problematyki i można chyba uważać, że zakończył się on w zasadzie w latach czterdziestych. Nie znaczy to oczywiście, że dziś nie prowadzi się prac nad rozstrzygalnością czy pełnością teorii matematycznych, podobnie zresztą nadal poszukuje się algorytmów rozwiązywania pewnych zadań — jednakże obecne badania w dziedzinie podstaw nie przyczyniają się do głębszego zrozumienia pojęcia algorytmu. Pojęcie algorytmu zostało sprecyzowane w obecnej postaci na początku okresu drugiego i do dnia dzisiejszego nie uległo ono zmianie.

Okres trzeci rozpoczął się w latach pięćdziesiątych wraz z powstaniem maszyn matematycznych. Końca tego okresu chwilowo nie widać. W związku z różnorodnym zastosowaniem maszyn matematycznych powstała olbrzymia liczba bardzo skomplikowanych algorytmów. Uważa się, że mieszczą się one w ogólnej koncepcji algorytmu podanej w związku z badaniami podstaw matematyki, chociaż nikt tego właściwie nie zbadał dokładnie i być może można by mieć wątpliwości odnośnie takiego stanowiska. Okres ten podobny jest więc do okresu pierwszego, gdyż chodzi tu również o poszukiwanie algorytmów, jednakże różnica między tymi okresami polega na tym, że obecnie nie chodzi o znalezienie jakiegokolwiek algorytmu, a algorytmu o specjalnych własnościach. Nie każdy algorytm jest jednakowo dobry do zastosowań maszynowych. Struktura tych algorytmów różni się zasadniczo od algorytmów poszukiwanych w pierwszym okresie. Obecne zastosowania maszyn zwróciły uwagę na strukturę algorytmów. O ile dla badacza podstaw stwierdzenie, że jakieś zagadnienie jest rozwiązywalne algorytmicznie jest zakończeniem pracy, o tyle z punktu widzenia zastosowań stwierdzenie tego faktu jest dopiero początkiem badań. Nie każdy algorytm jest jednakowo przydatny do celów maszynowych. Jednakże co to znaczy, że jeden algorytm jest „lepszy" od innego algorytmu — do tej pory nie bardzo wiadomo. Wydaje się, że aby na to pytanie odpowiedzieć konieczne jest znacznie głębsze wniknięcie w strukturę różnorodnych algorytmów, dostarczonych przez praktykę związaną z użytkowaniem współczesnych maszyn matematycznych.

Niewątpliwie w ciągu najbliższych kilku lat powstanie czwarty etap rozwoju pojęcia algorytmu, polegający na jakiejś syntezie olbrzymiego i różnorodnego materiału dostarczonego przez użytkowanie maszyn matematycznych oraz próbie stworzenia „nowej" teorii algorytmów badającej własności algorytmów o różnych strukturach z punktu widzenia ich realizowalności określonymi środkami. Także samo pojęcie algorytmu, w związku z obecnymi jego zastosowaniami, wymaga również ściślejszego określenia.

Jak pamiętamy uniwersalnym językiem matematyki, w którym można sformułować każdy problem matematyczny jest rachunek zdań i rachunek kwantyfikatorów. Przeprowadzanie rozumowań matematycznych sprowadza się do formalnego przekształcania wyrażeń, należących do obu tych rachunków. Język matematyki został oczywiście tak zbudowany, aby można było w nim wyrażać dowolne fakty matematyczne, jednakże jeżeli pominiemy stronę znaczeniową tego języka, to na teorie matematyczne możemy również spojrzeć jako na metody przekształcania symboli w myśl określonych reguł. Jakkolwiek taki punkt widzenia nie pozwala na głębsze wniknięcie w sens teorii matematycznych, to jednak niektóre problemy związane ze strukturą teorii matematycznych mogą być na tej drodze rozwiązane. Dlatego też matematycy zainteresowali się ogólnymi zasadami przekształcania symboli i badaniem niektórych własności takich systemów, mając między innymi na uwadze zastosowanie wyników uzyskanych na tej drodze do interesujących ich „prawdziwych" teorii matematycznych.

W rozdziale tym przedstawimy trzy najwcześniejsze historycznie systemy przekształcania symboli, powstałe w latach 1910-1943. Systemy te zostały podane przez Thuego, Posta i Markowa. Stanowią one do dzisiaj podstawę wielu kierunków badań związanych z teorią algorytmów, a ponadto próbuje się je zastosować do języków maszyn matematycznych, lingwistyki matematycznej i innych. Dlatego Czytelnik zainteresowany zastosowaniami logiki może zetknąć się łatwo z problematyką, którą zainicjowali Thue, Post i Markow. Stąd też zaczynamy część poświęconą pojęciu algorytmu od przedstawienia ich idei. Nie będziemy się jednakże interesowali stroną matematyczną tych zagadnień, a raczej zwrócimy uwagę na samą strukturę mechanizmu przekształcania symboli w tych systemach.

§ 50. ZAGADNIENIE SŁÓW

Problem, który przedstawimy w tym paragrafie, postawił w latach 1910-1914 matematyk norweski Axel Thue. Problem ten był badany szczegółowo, między innymi przez Markowa, Nowikowa [1955] i Cejtina. Z problemem tym zapoznaliśmy się już w § 49 w związku z omawianiem teorii półgrup. Spojrzymy jeszcze raz na ten problem z nieco innego punktu widzenia.

Alfabetem A będziemy nazywali skończony zbiór różnych symboli $a_1, a_2, \ldots, a_n$. Zgodnie z przyjętym sposobem oznaczania zbiorów zapiszemy

$$A = \{a_1, a_2, \ldots, a_n\}.$$

Elementy alfabetu będziemy nazywali *literami*. Każdy skończony ciąg liter

$$a_{i_1}, a_{i_2}, \ldots, a_{i_k}$$

należących do ustalonego alfabetu A nazwiemy *słowem* w alfabecie A. Słowa będziemy oznaczali P, Q, R, S, T itd. ewentualnie ze wskaźnikami u dołu. Wprowadzimy również pojęcie *słowa pustego*, tj. słowa nie zawierającego żadnego symbolu. Słowo puste oznaczymy symbolem $\varnothing$. Zbiór wszystkich słów w alfabecie A oznaczymy przez A^*. Liczbę symboli w słowie P nazwiemy *długością* słowa P i oznaczymy przez $d(P)$. Będziemy mówili że słowo P zawiera się w słowie Q, symbolicznie $P \subset Q$, jeżeli słowo Q ma postać RPS, przy czym słowa P, Q, R, S, należą do ustalonego alfabetu A. W szczególności słowa R i S mogą być puste.

Na przykład, przyjmijmy jako alfabet następujący zbiór liter

$$A_1 = \{a, b, c\}.$$

Słowami w alfabecie A_1 będą np. następujące ciągi liter

$$aaaa, \quad cabbc, \quad bbca.$$

Słowo aa zawiera się w słowie $aaaa$, trzy razy w $aacaaaa$, słowo zaś ab zawiera się w słowie $cabbc$,

Niech R będzie skończonym zbiorem par słów

$$\begin{aligned} &R_1: \quad X_1 - Y_1, \\ &R_2: \quad X_2 - Y_2, \\ &\qquad \ldots\ldots \\ &R_m: \quad X_m - Y_m, \end{aligned}$$

takich, że $X_i, Y_i \in A^*$. W szczególności X_i bądź Y_i może być słowem pustym.

Parę słów $X_i - Y_i$ będziemy rozumieli jako następującą regułę przepisywania słów: jeżeli słowo P ma postać RX_iS, to wolno nam w słowie P zawierające się w nim słowo X_i zastąpić słowem Y_i, otrzymując w ten sposób nowe słowo RY_iS. Podobnie w słowie RY_iS możemy zastąpić Y_i przez słowo X_i, otrzymując w rezultacie słowo RX_iS. Będziemy wtedy mówili, że reguła $X_i - Y_i$ jest stosowana do słowa P.

W ten sposób określiliśmy *zasady przekształcania słów*, pozwalające z jednych słów otrzymywać nowe słowa za pomocą ustalonych reguł przekształcania.

Zbiór A^* wraz ze zbiorem R będziemy nazywali *rachunkiem słów*. Rachunek słów jest więc określony przez podanie alfabetu oraz zbioru reguł przekształcania słów. W dalszym ciągu podamy prosty przykład rachunków słów.

PRZYKŁAD. Przyjmijmy następujący alfabet

$$A_1 = \{a, b, c\}$$

oraz trzy reguły przepisywania

$$\begin{aligned} &R_1: \quad aa - b, \\ &R_2: \quad ac - a, \\ &R_3: \quad aaa - c. \end{aligned}$$

Do słowa np. *abac* możemy zastosować regułę R_1, otrzymując słowo *aaaac*. Przebieg przekształcania słów będziemy zapisywać podobnie jak to czyniliśmy w dowodzeniu twierdzeń, pisząc słowa w kolejnych numerowanych wierszach i podając w każdym wierszu numer zastosowanej reguły przekształcenia oraz numer wiersza, w którym zapisano wynik przekształcenia:

1. *abac* $R_1, 2$;
2. *aaaac* $R_3, 3$;
3. *cac* $R_2, 4$;
4. *ca*.

W każdym słowie podkreślono słowo w nim zawarte, do którego zastosowano regułę przekształcania. Przekształcanie to moglibyśmy kontynuować dalej.

Jeżeli słowa P i Q mają odpowiednio postać RX_iS i UY_iW, oraz para $X_i — Y_i$ jest regułą rachunku słów, to słowa P i Q nazywamy *słowami sąsiednimi* i zapiszemy $P = Q$. Słowa T i S nazwiemy *równoważnymi* ze względu na zbiór reguł R i zapiszemy $T \sim S$, jeżeli istnieje taki ciąg słów

$$P_1, \quad P_2, \quad \ldots, \quad P_n,$$

że P_1 jest identyczne z T, a P_n jest identyczne z S i dla każdego i, $1 \leqslant i \leqslant n$, $P_i = P_{i+1}$. Ciąg $P_1, P_2, \ldots, P_n$ będziemy nazywali *wywodem słowa S* ze słowa T. Oczywiście jeżeli $P_1, P_2, \ldots, P_n$ jest wywodem słowa S ze słowa T, to $P_n, P_{n-1}, \ldots, P_1$ jest wywodem słowa T ze słowa S.

W podanym powyżej przykładzie wywód słowa *ca* ze słowa *abac* będzie miał postać

$$abac = aaaac = cac = ca.$$

A więc słowa *abac* i *ca* są równoważne ze względu na układ reguł R_1, R_2, R_3.

W rachunku słów istnieje następujący zasadniczy problem:

Zadany jest rachunek słów $\langle A, R\rangle$. Dla dowolnych dwu słów P, $Q \in A^*$ podać metodę rozstrzygania, czy słowa P i Q są równoważne ze względu na reguły R czy nie.

Jak wykazano problem ten ogólnie jest nierozstrzygalny, tj. nie można podać w ogólnym przypadku metody, która by pozwoliła stwierdzić, czy w tym rachunku dowolne dwa słowa są równoważne czy nie (innymi słowy nie można podać algorytmu rozwiązania) (porównaj § 49). Jednakże dla niektórych przypadków metoda taka istnieje. Rozpatrzymy jeden taki przypadek z punktu widzenia struktury algorytmu rozwiązania tego zadania.

Załóżmy mianowicie, że dla każdej reguły zachodzi zależność

$$d(X_i) = d(Y_i),$$

co znaczy, że w każdej regule liczba symboli w jednym i drugim słowie jest jednakowa (oczywiście w każdej regule może ona być inna). Wtedy problem słów jest rozstrzygalny z następującego powodu: jeżeli słowa P i Q są równoważne, to istnieje ciąg

$$P_0 = P_1 = P_2 = P_3 = \ldots = P_r = P_{r+1}, \tag{1}$$

w którym P_0 oznacza P, natomiast P_{r+1} oznacza Q. Ponieważ zastosowanie jakiejkolwiek reguły nie zwiększa liczby symboli w słowie, więc wszystkie słowa P mają jednakową długość.

Z uwagi na to, że można przyjąć iż, wszystkie słowa ciągu (1) są różne więc

$$r < l^k = M,$$

gdzie l — długość słowa P, a k — liczba symboli alfabetu. Wobec tego liczba ciągów, zaczynających się od P i kończących się na Q jest skończona. Tym samym problem słów dla tego przypadku jest rozstrzygalny. Mamy bowiem skończoną liczbę ciągów do przejrzenia i sprawdzenia, czy wśród nich znajduje się ciąg typu (1). A więc mamy metodę sprawdzenia, czy słowa P i Q są równoważne, czy też nie. Algorytm tego postępowania moglibyśmy więc zapisać w postaci wskazówek:

Wskazówka 1. Wypisz wszystkie możliwe ciągi różnych słów literowych postaci

$$\begin{aligned} &P, Q, \\ &P, P_1, Q, \\ &P, P_1, P_2, Q, \\ &\ldots\ldots\ldots\ldots \\ &P, P_1, P_2, \ldots, P_r, Q. \end{aligned} \tag{2}$$

Wskazówka 2. Sprawdź, czy wśród wypisanych ciągów istnieje ciąg dla którego zachodzi

$$P = P_1 = P_2 = \ldots = P_r = Q, \quad 0 \leqslant r \leqslant M. \tag{3}$$

Wskazówka 3. Jeżeli wśród ciągów (2) istnieje ciąg postaci (3), to daj odpowiedź „słowa P i Q są równoważne" — jeżeli wśród ciągów (2) nie ma ciągu postaci (3), to daj odpowiedź „słowa P i Q nie są równoważne".

Wskazówki 1, 2 i 3 dotyczą czysto formalnych manipulacji na symbolach. Do ich wykonania nie potrzeba rozumieć problemu słów. Aby więc rozwiązać problem słów w tym przypadku, możemy kogoś pouczyć w jaki sposób ma wypisać wszystkie ciągi typu (2), jak ma następnie sprawdzać, czy wśród nich znajduje się ciąg postaci (3) i zależnie od otrzymanego wyniku, jaką ta osoba ma dać odpowiedź. Każdą ze wskazówek 1, 2, 3 można jeszcze dokładniej rozbić na szereg bardziej szczegółowych wskazówek tak, że cały przebieg postępowania będzie zupełnie jasny dla każ-

dego. Taki właśnie przepis postępowania mamy na myśli, mówiąc o algorytmie rozwiązywania jakiegoś zagadnienia. Do postępowania według algorytmu nie trzeba rozumieć treści rozwiązywanego zagadnienia, wystarczy tylko ściśle wypełniać podane w algorytmie wskazówki.

Gdybyśmy chcieli rzeczywiście w podany sposób rozstrzygać, czy słowa są równoważne, czy nie, musielibyśmy się solidnie napracować, a często praktycznie nie bylibyśmy w stanie w jakimś rozsądnym czasie wykonać tak dużej liczby operacji na symbolach, jaka jest wymagana dla rozwiązania tego zagadnienia. Są to liczby niemal astronomiczne (patrz zad. 1 tego paragrafu). Podany algorytm jest więc dobry, gdy chcemy stwierdzić potencjalną możliwość sprawdzenia równoważności słów, natomiast gdybyśmy chcieli rzeczywiście algorytm ten zrealizować, zadanie to przy podanym algorytmie będzie na ogół praktycznie niewykonalne, nawet gdybyśmy do niego zastosowali nowoczesne szybkie maszyny matematyczne.

W różnych teoriach matematycznych często zachodzi potrzeba stwierdzenia, czy jakiś problem jest rozstrzygalny czy nie. Wystarczy wtedy udowodnić istnienie odpowiedniego algorytmu, nie zajmując się bliżej jego możliwością praktycznego wykonania.

W zastosowaniach logiki, szczególnie w maszynych matematycznych i w teorii języków, również występuje bardzo często konieczność rozwiązania problemu słów dla zadanego alfabetu i zbioru reguł przekształcania, jednakże nie chodzi nam wtedy tylko o stwierdzenie, czy problem jest rozstrzygalny, ale również o znalezienie algorytmu rozstrzygania w takiej postaci, aby był on realizowalny środkami, którymi dysponujemy.

Dla rozważanego tu przykładu można podać znacznie prostszy algorytm postępowania, np. taki: dla zadanego słowa P należy utworzyć zbiór wszystkich słów sąsiednich (jest ich oczywiście liczba skończona) i sprawdzić, czy istnieje wśród nich słowo Q. Jeżeli nie istnieje, dla każdego słowa sąsiedniego P znaleźć zbiór słów sąsiednich itd. Ponieważ liczba różnych słów, które możemy w ten sposób otrzymać jest skończona, wcześniej czy później natrafimy na słowo Q. Ten algorytm jest znacznie prostszy od poprzedniego, wymaga znacznie mniej pracy i liczby operacji, które należy wykonać dla sprawdzenia równoważności słów P i Q. Należy zwrócić uwagę, aby tworząc zbiór słów sąsiednich dla danego słowa nie wypisywać słów, które zostały otrzymane już poprzednio, gdyż w przeciwnym przypadku moglibyśmy procesu tworzenia nowych słów w ogóle nie

skończyć, powtarzając cyklicznie te same czynności. Dochodzi więc tutaj dodatkowy kłopot polegający na tym, że po otrzymaniu każdego nowego słowa musimy sprawdzić, czy w trakcie dotychczasowego postępowania nie zostało ono już utworzone. W sumie algorytm ten jest jednak prostszy od algorytmu poprzedniego. Można się zastanawiać, czy nie istnieją jeszcze inne algorytmy, jeszcze bardziej przydatne do praktycznej realizacji. Zagadnienie upraszczania algorytmów jest jednym z głównych problemów teorii maszyn matematycznych.

Streszczenie

Problem słów polega na podaniu metody rozstrzygania, czy dowolne dwa słowa są równoważne ze względu na zadany układ reguł przekształcania słów, tzn. czy jedno z nich można otrzymać z drugiego przez przekształcanie za pomocą odpowiednich reguł. W ogólnym przypadku problem ten jest nierozstrzygalny, natomiast przy specjalnych założeniach upraszczających algorytm rozstrzygania dla problemu słów istnieje.

Zadania

1. a) Obliczyć, posługując się wzorami podanymi w § 31 z rozdziału V, ile w alfabecie złożonym z n liter jest ciągów o długości $\leqslant d$.

b) Obliczyć, ile różnych układów ciągów $P, Q, P, P_1, Q, \ldots, P, P_1, \ldots, P_M, Q$, gdzie $d(P) = d(Q) = d(P_i) = d$ można wypisać, w zależności od:

liczby liter w alfabecie,

długości d,

liczby składników równej $M+2$.

2. Przyjmując reguły $ab—ba$, $ac—ca$, $bc—ca$ zbadać czy słowa

a) $abac$ i $bcac$,

b) abc i abb

są równoważne.

Wypisać wszystkie słowa pośrednie użyte w tym sprawdzeniu.

§ 51. JĘZYKI POSTA

Obecnie zapoznamy się z nieco innymi systemami przekształcania symboli podanymi przez matematyka amerykańskiego Emila Posta [1943], [1946], [1947], w latach 1920-1943. Systemy te nazywamy *językami Posta.*

Języki Posta są również dzisiaj intensywnie studiowane w związku z teorią maszyn matematycznych, podobnie jak systemy Thuego. Post rozpatrywał trzy rodzaje języków, które kolejno omówimy.

1. Języki kanoniczne Posta. Podobnie jak w poprzednim paragrafie, A będzie skończonym alfabetem, a skończone ciągi symboli tego alfabetu będą słowami, które będziemy podobnie jak poprzednio oznaczali literami P, Q, R, S, itp. W zbiorze słów A^* wyróżnimy skończony podzbiór B wyrażeń, które nazwiemy *wyrażeniami pierwotnymi* (lub *aksjomatami*). Zbiór B zawiera również ciąg pusty $\varnothing$.

Językiem kanonicznym Posta będziemy nazywali układ $\langle A, B, \mathrm{R}\rangle$, gdzie A jest alfabetem, B — zbiorem wyrażeń pierwotnych, R zaś jest skończonym zbiorem reguł przekształcania symboli postaci:

$$(1)\qquad \frac{\begin{array}{l} A_{11}X_{11}A_{12}X_{12}\ldots A_{1n_1}X_{1n_1}A_{1n_1+1} \\ A_{21}X_{21}A_{22}X_{22}\ldots A_{2n_2}X_{2n_2}A_{2n_2+1} \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ A_{k1}X_{k1}A_{k2}X_{k2}\ldots A_{kn_k}X_{kn_k}A_{kn_k+1} \end{array}}{A_1X_1A_2X_2\ldots A_rX_rA_{r+1}}$$

Reguły typu (1) są nazywane *produkcjami*. Litery A ze wskaźnikami oznaczają dowolne wyrażenia pierwotne, $A_{ij}, A_i \in B$, natomiast litery X ze wskaźnikami oznaczają dowolne skończone ciągi symboli należące do zbioru A^*-B. Elementy zbioru A^*-B są nazywane *danymi produkcji* (albo *przesłankami produkcji*), natomiast ciąg pod kreską w wyrażeniu (1) jest nazywany *bezpośrednim produktem danych* (albo *bezpośrednim wnioskiem z przesłanek*). Zakładamy ponadto, że każda zmienna X_i występująca w produkcie bezpośrednim, występuje chociaż w jednej z danych produkcji — oraz, że każda z danych oraz produkt bezpośredni zawierają choć jedną zmienną.

Przykład 1. Przykład języka kanonicznego. Jako alfabet przyjmiemy

$$A_1 = \{a, b, c\},$$

jako zbiór zaś wyrażeń pierwotnych, następujące wyrażenia

$$B_1: \quad ab, \quad B_2: \quad aa, \quad B_3: \quad ac.$$

Jako jedyną produkcję w tym przykładzie weźmiemy regułę

$$\mathrm{R}_1: \qquad \frac{\begin{array}{l} B_1XB_2 \\ B_1XB_3 \end{array}}{B_2XB_3}$$

Regułę R_1 możemy też napisać w postaci

$$\mathrm{R}_1: \qquad \left(\frac{\begin{array}{l} abXaa \\ abXac \end{array}}{aaXac}\right)$$

Reguła ta znaczy, że jeżeli mamy dwa ciągi postaci *abXaa* oraz *abXac*, gdzie X oznacza dowolny ciąg liter a, b, c z wyjątkiem ciągów, *ab*, *aa* oraz *ac*, to zamiast obu danych możemy napisać ciąg *aaXac*. Na przykład

$$\frac{\begin{array}{l} \underline{abccabcaa} \\ \underline{abccabcac} \end{array}}{\underline{aaccabcac}}$$

Podkreślone ciągi są wyrażeniami pierwotnymi. Produkcja pozwala więc tworzyć z danych ciągów symboli nowe ciągi.

Wszystkie wyrażenia, które możemy otrzymać z wyrażeń pierwotnych przez wielokrotne stosowanie produkcji dopuszczalnych w języku — będziemy nazywali *twierdzeniami języka*. Mówiąc nieco dokładniej, ciąg słów

$$P_1, P_2, \ldots, P_n$$

nazwiemy *dowodem ciągu* P_n w języku J, jeżeli dla każdego i $(1 < i \leqslant n)$ P_i jest dane, albo istnieją takie ciągi $P_{i_1}, P_{i_2}, \ldots, P_{i_k}$, $i_j < i$, że P_i jest bezpośrednią produkcją ciągów $P_{i_1}, P_{i_2}, \ldots, P_{i_k}$. To znaczy przyjmując jako słowo wyjściowe wyrażenia pierwotne języka oraz dobierając na zmienne odpowiednie słowa języka musimy za pomocą dopuszczalnych produkcji wyprodukować twierdzenie, o które nam chodzi.

Język Posta ma charakter uniwersalny. Zależnie od przyjętego alfabetu oraz produkcji i wyrażeń pierwotnych możemy otrzymać różne języki matematyczne. Dla języków kanonicznych Posta istnieje następujący problem: dla dowolnego języka kanonicznego $\langle A, B, \mathrm{R}\rangle$ i dla słowa należącego do A^* podać metodę rozstrzygania, czy słowo to jest twierdzeniem w tym języku, czy nie.

Problem ten w ogólnym przypadku jest nierozstrzygalny. Wprawdzie można, posługując się produkcjami, produkować kolejno wszystkie możliwe twierdzenia dla każdego języka kanonicznego Posta, jednakże nie istnieje algorytm pozwalający znaleźć dowód dla z góry zadanego słowa tego języka lub stwierdzić, że słowo nie jest twierdzeniem. Z punktu widzenia zastosowań, interesujące są takie języki kanoniczne Posta, dla których istnieje algorytm rozstrzygania, czy dowolne słowo jest twierdzeniem czy nie. W takim przypadku pojawia się, podobnie jak i w systemie Thuego, problem struktury algorytmu rozstrzygania, tj. poszukiwanie algorytmów efektywnie wykonywanych określonymi środkami. W ogólnym przypadku bowiem dla języków rozstrzygalnych Posta, poszukiwanie dowodu sprowadza się również do sprawdzenia skończonej liczby ciągów słów, czy są one dowodami czy nie dla zadanego słowa. Jednakże w niektórych przypadkach języków rozstrzygalnych, dowody te można znaleźć w prostszy sposób, bez konieczności rozpatrywania bardzo dużej liczby przypadków. Szukanie takich języków jest ważne z punktu widzenia zastosowań.

2. Języki normalne Posta. Post wykazał, że każdy język kanoniczny może być zredukowany do języka normalnego, który jest także językiem kanonicznym specjalnej postaci. Znaczy to, że dla każdego języka kanonicznego można dobrać taki język normalny, że każde twierdzenie ustalonego języka kanonicznego jest również twierdzeniem odpowiedniego języka normalnego i odwrotnie. *Językiem normalnym Posta* będziemy nazywali taki język kanoniczny, w którym istnieje co najmniej jedno wyrażenie pierwotne i każda produkcja ma postać

$$\frac{A_1 X}{X A_2}$$

gdzie A_1 i A_2 są wyrażeniami pierwotnymi oraz X jest dowolnym słowem różnym od A_1 i A_2. Zamiast pisać produkcję w postaci jak wyżej, wygodniej jest w tym przypadku przyjąć następujący sposób zapisu:

$$A_1 X \to X A_2 .$$

W ten sposób problem rozstrzygalności dla języków kanonicznych jest sprowadzony do problemu rozstrzygalności dla języków normalnych. Warto zauważyć, że produkcje w językach normalnych Posta są o wiele prostsze od produkcji w językach kanonicznych. Gdybyśmy więc rzeczywiście chcieli realizować jakieś algorytmy, a nie interesowali się oboma językami tylko z punktu widzenia rozstrzygalności, to języki normalne są niewątpliwie bardziej interesujące od języków kanonicznych. Rozpatrzymy prosty przykład języka normalnego.

PRZYKŁAD 2. Niech alfabet składa się tylko z dwu liter a i b:

$$A = \{a, b\}.$$

Jako wyrażenia pierwotne przyjmiemy

$$B_1: \quad aa,$$
$$B_2: \quad bb.$$

Produkcją w tym języku będzie reguła

$$\mathrm{R}_1: \quad aaX \to Xbb.$$

Sprawdźmy, czy słowo *aaaabbbb* jest twierdzeniem rozpatrywanego języka normalnego. Aby było ono twierdzeniem, zgodnie z definicją podaną w poprzednim ustępie, musimy umieć je wyprodukować z wyrażeń pierwotnych *aa* oraz *bb* za pomocą reguły R_1. Ponieważ można podać następujący dowód

1. *aaaaaaaa* $\mathrm{R}_1, 2,$
2. *aaaaaabb* $\mathrm{R}_1, 3,$
3. *aaaabbbb*,

więc słowo *aaaabbbb* jest twierdzeniem w tym języku normalnym.

3. Tag-system Posta(1).

(1) tag oznacza w języku angielskim gonitwę, zabawę w berka. Ponieważ brak nam odpowiedniego słowa w języku polskim, które by oddawało dobrze sens słowa tag w odniesieniu do rozpatrywanych problemów — pozostaniemy przy terminie angielskim, tym bardziej że jest powszechnie używany w literaturze światowej dotyczącej poruszanego tematu.

W związku z badaniami systemów normalnych Post zdefiniował szczególny rodzaj systemu normalnego, który jest określony następująco:

Dany jest skończony alfabet $A = \{a_0, a_1, \ldots, a_n\}$ oraz liczba naturalna k. Każdemu symbolowi $a_i \in A$ jest przyporządkowane słowo $P_i \in A^*$. P_i może być w szczególności słowem pustym $\varnothing$.

$$\begin{aligned} &a_0 \to P_0, \\ &a_1 \to P_1, \\ &\ldots\ldots \\ &a_n \to P_n. \end{aligned} \tag{2}$$

Wprowadzamy następnie jedną operację na słowach języka oznaczoną przez T. Jeżeli $Q = a_{i_1}, a_{i_2}, \ldots, a_{i_n}$ jest dowolnym słowem należącym do A^*, to $T(Q)$ jest słowem otrzymanym następująco: na końcu słowa Q dopisujemy ciąg P_{i_1} odpowiadający pierwszemu symbolowi słowa Q i następnie z tak otrzymanego słowa wymazujemy pierwszych k liter, otrzymując w ten sposób słowo

$$a_{i_{k+1}}, \quad \ldots, \quad a_{i_n}, \quad P_{i_1}, \tag{3}$$

które jest rezultatem naszej operacji. Do otrzymanego słowa (3) możemy również zastosować operację T, itd. Proces uważamy za zakończony, jeżeli w rezultacie stosowania operacji T otrzymamy ciąg pusty.

PRZYKŁAD. Przyjmijmy alfabet $A = \{a, b, c\}$ oraz następujące przyporządkowanie

$$\begin{aligned} &a \to aba, \\ &b \to cbb, \\ &c \to b. \end{aligned}$$

Niech ponadto $k = 2$.

Zastosujemy operacje T do ciągu *baccab*:

1. *baccab*,
2. *ccabcbb*,
3. *abcbbb*,
4. *cbbbaba*,
5. *bbabab*.

Operacje te możemy stosować dalej jednakże nie wiemy, czy proces ten się zakończy, czy nie. W związku z powyższym Post postawił następujące dwa problemy:

a) Podać algorytm rozstrzygający, czy dla zadanego układu przyporządkowań l, zadanej liczby k oraz dla zadanego słowa proces stosowania operacji T do tego słowa kończy się czy nie.

b) Podać algorytm rozstrzygający, czy zadane słowo P można otrzymać ze słowa Q, stosując skończoną ilość razy operację T do słowa Q.

Oba te problemy okazały się bardzo trudne i do tej pory nie znaleziono ich ogólnego rozwiązania. System ten jest jednakże również interesujący nie tylko z punktu widzenia rozstrzygalności, a także z punktu widzenia techniki przekształcania symboli i może on stanowić interesujący materiał dla studiów nad praktyczną efektywnością algorytmów.

Streszczenie

Określiliśmy trzy rodzaje języków podanych przez Posta — języki kanoniczne, języki normalne oraz języki zwane tag-systemami. Wszystkie te języki zawierały reguły przekształcania słów pozwalające z jednych słów otrzymywać nowe słowa. Zasadniczy problem dla tych języków polegał na podaniu algorytmu rozstrzygania, czy zadane słowo można wyprowadzić z pewnych słów początkowych za pomocą dopuszczalnych reguł przekształceń. Problem ten w ogólnym przypadku jest nierozstrzygalny dla wszystkich trzech języków. Z drugiej strony reguły przekształceń wyrażeń w tych językach mają charakter na tyle ogólny, że mieszczą się w nich prawie wszystkie języki matematyczne.

Zadania

1. a) Podać wyprowadzenie z wyrażeń pierwotnych i reguł podanych w przykładzie 1, następujących słów:

ab *ccabcaa*

Wypisać po kolei wszystkie kroki wyprowadzenia.

b) uzasadnić, że słowo

ba ccabcaa

różniące się od słowa z punktu a) porządkiem dwóch pierwszych symboli nie może być wyprowadzone z podanych w przykładzie 1 reguł.

c) O jakie wyrażenie należy wzbogacić zbiór wyrażeń B_1-B_3 podanych w tym przykładzie, by słowo to dało się wyprowadzić? Dodać to wyrażenie pierwotne do wyrażeń B_1-B_3 i napisać pełny wywód słowa.

d) Podać prostą cechę odróżniającą słowa dające się wyprowadzić z reguły R_1 (z przykładu 1) od słów nie dających się wyprowadzić za pomocą tej reguły z wyrażeń pierwotnych B_1-B_3. Uzasadnić, że podany w tym przykładzie system jest rozstrzygalny.

Uwaga. Nie każdy system kanoniczny Posta jest rozstrzygalny.

2. a) Czy słowo *aaabbbbb* jest twierdzeniem w systemie normalnym, podanym w przykładzie 2?

b) Czy słowo *aabbbb* jest twierdzeniem w tym systemie?

c) Uzasadnić, że system ten jest rozstrzygalny i podać prostą cechę odróżniającą te słowa, które są twierdzeniem od tych, które nie są twierdzeniami.

Uwaga. Nie każdy system normalny Posta jest rozstrzygalny.

§ 52. ALGORYTMY NORMALNE MARKOWA

Jak można to było zauważyć z przykładów podanych w poprzednich paragrafach, algorytmy rozstrzygania czy słowo daje się uzyskać w systemie, czy też nie, nie były jednoznacznie określone. Każdy z podanych problemów można było rozwiązać na wiele sposobów, podając różne sposoby postępowania, tj. różne algorytmy, jeżeli problemy te były oczywiście rozstrzygalne. W tym paragrafie zapoznamy się z pojęciem algorytmu wprowadzonym przez matematyka rosyjskiego — Markowa, który od jego nazwiska jest nazywany normalnym algorytmem Markowa. Algorytmy Markowa można uważać również za zespoły reguł przekształcania symboli, z tą różnicą w stosunku do poprzednich koncepcji, że obecnie sposób i kolejność posługiwania się regułami są ściśle określone i nie można ich stosować w dowolnym porządku, tak aby otrzymać zamierzony wynik.

Niech A będzie alfabetem i niech A' będzie alfabetem nie zawierającym symboli należących do A. Symbole należące do A' nazwiemy *symbolami pomocniczymi* a symbole należące do A — *symbolami podstawowymi*.

Normalnym algorytmem Markowa nazwiemy ponumerowany ciąg reguł postaci:

$$\begin{aligned} &1. \quad X_1 \to Y_1,\\ &2. \quad X_2 \to Y_2,\\ &\quad \dots\dots\dots\\ &n. \quad X_n \to Y_n. \end{aligned}$$

Nad każdą ze strzałek może być ewentualnie kropka. $X_1, X_2, \dots, X_n$, $Y_1, Y_2, \dots, Y_n$ są słowami należącymi do $(A \cup A')^*$[1] (dopuszczalne są również słowa puste). Sens strzałki $\to$ jest taki sam, jak w poprzednich paragrafach. Reguły zawierające strzałki z kropką będziemy nazywali *regułami końcowymi*, pozostałe zaś — *regułami niekońcowymi*. Jeżeli słowo X_i zawiera się w słowie P, to powiemy, że *reguła* nr i jest *stosowalna* do słowa P. Jeżeli reguła nr i jest stosowalna do słowa P, to możemy w słowie P pierwszy raz występujące (licząc od lewej) słowo X_i zastąpić słowem Y_i. Słowo X_i może bowiem w słowie P występować wiele razy. Na przykład P może mieć postać $P_1 X_i P_2 X_i P_3$. Wtedy regułę nr i możemy zastosować tylko do pierwszego od lewej strony słowa X_i.

Stosowanie algorytmu do zadanego słowa jest następujące: najpierw do zadanego słowa próbujemy stosować regułę nr 1. Jeżeli to jest niemożliwe — to regułę nr 2 itd, aż do znalezienia reguły, która jest do zadanego słowa stosowalna. Jeżeli takiej reguły nie znaleźliśmy, to proces jest zakończony. Jeżeli zaś znaleźliśmy regułę stosowalną do słowa P, która jest regułą końcową, to również kończymy proces stosowania algorytmu. Jeżeli natomiast zastosowana reguła nie jest regułą końcową, to ze słowa początkowego otrzymamy nowe słowo, do którego próbujemy ponownie stosować regułę nr 1 itd. — i postępujemy identycznie, jak dla słowa pierwszego. Proces stosowania reguł algorytmu powtarzamy tak długo, jak to jest możliwe, tj. do momentu aż nie natrafimy na regułę końcową, bądź też żadna z reguł nie jest już do otrzymanego słowa stosowalna. Ostatnie słowo takiego ciągu nazywamy *wynikiem algorytmu*.

Jeżeli wszystkie słowa w regułach algorytmu należą tylko do A^*, to algorytm nazywamy *algorytmem w alfabecie* A, jeżeli natomiast słowa

---

[1]) Przez B^* tutaj, tak jak i w poprzednim paragrafie, rozumiemy zbiór wszystkich słów zbudowanych w alfabecie B.

należą do $(A \cup A')^*$, to *algorytm* nazywamy *nad alfabetem A*. Rozpatrzmy proste przykłady algorytmów w alfabecie i nad alfabetem.

PRZYKŁAD 1. Niech alfabet A składa się z trzech liter a, b, c, natomiast algorytm niech ma postać:

1. $ab \to ca$,
2. $aa \to b$,
3. $cc \to a$.

Zastosujmy ten algorytm do słowa *abaacab*

abaacab	1
caaacab	1
caaacca	2
cbacca	3
cbaaa	2
cbba	

Z prawej strony każdego słowa podano numer zastosowanej reguły algorytmu. Do ostatniego słowa nie możemy już zastosować żadnej reguły algorytmu, jest ono więc wynikiem algorytmu zastosowanego do słowa *abaacab*. Był to oczywiście algorytm w alfabecie A.

PRZYKŁAD 2. Podamy teraz przykład algorytmu nad alfabetem A (zaczerpnięty z Curry'ego), tworzenia ze słowa $P \in A^*$ słowa PP. Jako alfabet podstawowy przyjmiemy alfabet A z poprzedniego przykładu. Jako symbole pomocnicze przyjmiemy litery x i y. Algorytm ten będzie miał postać:

1. $xa \to ayax$,
2. $xb \to bybx$,
3. $xc \to cycx$,
4. $yaa \to aya$,
5. $yab \to bya$,
6. $yac \to cya$,
7. $yba \to ayb$,
8. $ybb \to byb$,
9. $ybc \to cyb$,
10. $yca \to ayc$,

11. $ycb \to byc$,
12. $ycc \to cyc$,
13. $y \to \varnothing$,
14. $x \to \varnothing$,
15. $\varnothing \to x$.

Zobaczymy teraz w jaki sposób algorytm ten zastosowany do jakiegoś słowa P produkuje słowo PP. Jako słowo P weźmy *baac*. Możemy przyjąć, że słowo to zaczyna się i kończy symbolem pustym $\varnothing$. Z prawej strony każdego słowa podamy numer zastosowanej reguły algorytmu. Najpierw podamy cały przebieg algorytmu, a potem wyjaśnimy jego działanie.

I	$\varnothing baac$	15
	$xbaac$	2
	$bybxaac$	1
	$bybayaxac$	1
	$bybayaayaxc$	3
	$bybayaayacycx$	7
	. . . . .	
II	$baybyaayacycx$	4
	$baybayayacycx$	7
	$baaybyayacycx$	6
	$baaybyacyayacx$	6
	$baaybcyayayacx$	9
	$baacybyayayacx$	13
	. . . . .	
III	$baacbyayayacx\varnothing$	13
	$baacbayaycx\varnothing$	13
	$baacbaaycx\varnothing$	13
	$baacbaacx\varnothing$	14
	$baacbaac$	

Proces ten składa się z trzech części, oznaczonych I, II, III. W części I ze słowa *baac* utworzone zostało słowo *bbaaaacc* z odpowiednio porozmieszczanymi w nim symbolami pomocniczymi y i x. W części II

wszystkie podwójne litery, należące do alfabetu A, zostały przeniesione na koniec słowa i w drugiej części słowa są one poprzedzielane symbolem pomocniczym y. W ostatniej trzeciej części usunięte zostają wszystkie litery pomocnicze. (Regułę 13 i 14 pozwalającą na zastąpienie symbolu pomocniczego słowem pustym $\varnothing$ należy rozumieć jako po prostu wymazanie symbolu pomocniczego x bądź y). Ponieważ reguła 14 jest końcowa, więc proces przekształcania słów jest zakończony.

Dodatkowych wyjaśnień wymaga jeszcze pierwszy krok algorytmu. Mianowicie do słowa *baac* można jedynie zastosować regułę 15, mówiącą, że symbol pusty na początku słowa możemy zastąpić symbolem pomocniczym x. W ten sposób do słowa nie zawierającego symboli pomocniczych wprowadzamy pierwszy symbol pomocniczy i dalej możemy już posługiwać się regułami algorytmu.

Zwróćmy jeszcze uwagę, że gdybyśmy zmienili porządek reguł w algorytmie, sens jego uległ by zmianie. Reguły algorytmu należy bowiem stosować w takiej kolejności do słów, w jakiej występują one w algorytmie. W systemach Posta i Thue'go porządek reguł nie był ustalony i mogliśmy je zastosować w dowolnej kolejności. Stąd właśnie brała się niejednoznaczność poprzednich systemów.

Streszczenie

Algorytmy normalne Markowa są zbiorem ponumerowanych reguł, pozwalających na przekształcanie słów w jakimś alfabecie. Reguły stosujemy w kolejności ich występowania w algorytmie, w ten sposób, że w przekształcanym słowie P szukamy najpierw słowa X_1, poczynając od lewej strony słowa P. Jeżeli je znaleźliśmy, to zastępujemy je słowem X_1, jeżeli nie, to szukamy w słowie P słowa X_2 itd. Algorytm przestajemy stosować, jeżeli napotkamy na regułę końcową lub do otrzymanego słowa nie możemy zastosować żadnej reguły algorytmu.

Zadania

1. a) Podać, opierając się na przykładzie 2, algorytm tworzenia ze słowa $P \in A^*$ słowa PPP. Uogólnić.

b) Podać algorytm tworzenia ze słowa $P \in A^*$ słowa $\bar{P}$ zawierającego symbole P napisane w odwrotnym porządku.

c) Niech Q będzie dowolnym słowem w alfabecie A nie zawierającym litery q.

Podać algorytm tworzący z dowolnego słowa $P \in A^*$ słowo, w którym w każdym miejscu wstawiono Q zamiast litery q.

2. Opisać takie słowa o długości $\leqslant 3$, które nie ulegają zmianie, gdy zastosujemy do nich algorytm podany w przykładzie 1.

3. a) Alfabet A składa się z symboli 0 i 1. Słowa w tym alfabecie przedstawiają liczby naturalne (gdyż są rozwinięciami dwójkowymi liczb naturalnych). Podać algorytm, który z dowolnego słowa P należącego do A^* przedstawiającego liczbę n tworzy słowo $S(R)$ przedstawiające liczbę $n+1$.

b) Podać algorytm, który eliminuje zbędne symbole 0 (zero) stojące na końcu słowa.

§ 53. JĘZYKI CHOMSKY'EGO

Na zakończenie tego rozdziału podamy jeszcze jedną klasę języków, związaną z badaniem gramatyk języków naturalnych (jakkolwiek może mieć ona również zastosowanie do języków matematycznych) — nazwaną od nazwiska twórcy tego kierunku. Sam Chomsky języki, o których będzie mowa w tym paragrafie, nazwał *językami prostych struktur frazowych* (albo językami bezkontekstowymi) — w odróżnieniu od innych języków.

Teoria, której celem jest stosowanie metod matematycznych w językoznawstwie, jest nazywana *lingwistyką matematyczną*. Jednym z głównych zadań lingwistyki matematycznej jest podanie algorytmów rozstrzygania, czy zadane zdanie należy do określonego języka czy też nie. Stąd też problematyka lingwistyki matematycznej jest bardzo blisko związana z zagadnieniami podstaw matematyki a w szczególności z problemami rozstrzygalności. Ponieważ jednocześnie, niezależnie od czysto teoretycznej problematyki występującej w tej dziedzinie wiedzy, jednym z celów lingwistyki matematycznej jest podawanie praktycznych algorytmów rozstrzygania przynależności zdań do języka, gdyż służą one jako podstawa maszynowego tłumaczenia języków — lingwistyka dostarcza wielu interesujących przykładów algorytmów.

Koncepcja języka podana przez Chomsky'ego jest jedną z wielu możliwości formalnego ujęcia problemów gramatycznych. Innych jednak koncepcji nie będziemy tu podawali, a przedstawimy jedynie jedną z idei

Chomsky'go, gdyż wiąże się ona blisko z zagadnieniami poruszanymi w poprzednich paragrafach tego rozdziału. Nie będziemy również zajmowali się zastosowaniami tych idei do językoznawstwa, odsyłając Czytelnika bliżej zainteresowanego tym problemem do literatury fachowej (patrz Chomsky, Bar Hillel), a potraktujemy podany w tym paragrafie przykład języka, jako jeszcze jedną z metod przekształcania symboli.

Niech S będzie skończonym zbiorem symboli $s_1, s_2, \ldots, s_k, p_1, p_2, \ldots, p_r$. Elementy zbioru S będziemy nazywali słowami. Słowa $s_1, s_2, \ldots, s_k$ nazwiemy *słowami końcowymi słownika* S a zbiór słów końcowych słownika S oznaczymy przez K_s. Słowa $p_1, p_2, \ldots, p_r$ nazwiemy *słowami pomocniczymi słownika* S. Wśród słów pomocniczych słownika S wyróżnimy jedno słowo p_1, które nazwiemy słowem początkowym i oznaczymy je literą p. Przez S^* oznaczymy zbiór wszystkich skończonych ciągów słów należących do S. Elementy zbioru S^* będziemy nazywali zdaniami języka. Zdania będziemy oznaczali dużymi literami łacińskimi a słowa małymi literami. *Gramatyką języka* będziemy nazywali skończony ciąg reguł postaci

$$(1)\qquad \begin{aligned} P_{i_1} &\to A_1,\\ P_{i_2} &\to A_2,\\ &\cdots\cdots\\ P_{i_n} &\to A_n, \end{aligned}$$

gdzie $P_{i_j} \in P_S$, $A_i \in S,^*$ oraz $P_{i_j} \neq A_j$.

Reguły (1) należy rozumieć jak reguły z poprzednich paragrafów, tj. jako reguły przepisywania, pozwalające zastąpić w dowolnym zdaniu $Q \in S^*$ jeden symbol P_{i_j} odpowiadającym mu zdaniem A_j.

Językiem J będziemy nazywali parę $\langle S, G\rangle$, gdzie S jest słownikiem języka J, a G — gramatyką języka J.

Powiemy, że zdanie P wynika bezpośrednio ze zdania Q w gramatyce G i zapiszemy $Q \underset{G}{\Rightarrow} P$, lub krócej $Q \Rightarrow P$, jeżeli istnieją takie zdania X i Y, że zdanie P ma postać $XP_{i_j}Y$ zdanie Q zaś ma postać XA_jY, oraz wyrażenie $P_{i_j} \to A_j$ jest regułą gramatyki G.

Powiemy, że zdanie P wynika ze zdania Q w gramatyce G, symbolicznie $Q \to P$, jeżeli istnieją takie zdania

$$X_0, X_1, \ldots, X_s,$$

że P jest identyczne z X_0, Q jest identyczne z X_s oraz dla każdego i, $0 \leqslant i \leqslant n-1$, $X_i \Rightarrow X_{i+1}$.

Ciąg zdań

$$X_0, X_1, \ldots, X_s,$$

nazwiemy *wywodem* zdania P ze zdania Q w gramatyce S.

Zdanie, którego wszystkie słowa są końcowe nazwiemy *zdaniem końcowym*. Zdanie końcowe P nazwiemy *zdaniem poprawnym* w języku $J = \langle S, G\rangle$, jeżeli w gramatyce istnieje wywód zdania P z symbolu początkowego p.

Rozpatrzmy teraz przykład języka Chomsky'ego. Niech słownik języka J_1 składa się z następujących słów a, b, c, p, x, y. Słowa a, b, c, są słowami końcowymi p, x, y — słowami pomocniczymi. Słowo p jest słowem początkowym. Gramatyką słownika J_1 będzie następujący zespół reguł:

$$\begin{aligned} &R_1: && p \to xab,\\ &R_2: && x \to xy,\\ &R_3: && x \to c,\\ &R_4: && y \to ay,\\ &R_5: && y \to ac. \end{aligned}$$

Sprawdźmy obecnie czy zdanie *caacacab* jest zdaniem poprawnym języka J_1.

1.	p	R_1
2.	$\underline{xab}$	R_2
3.	$\underline{xy}ab$	R_2
4.	$\underline{xy}yab$	R_3
5.	$\underline{cy}yab$	R_4
6.	$\underline{ca}yyab$	R_5
7.	$\underline{caac}yab$	R_5
8.	$caacacab$	

Podobnie jak w poprzednich paragrafach, z prawej strony podano zastosowane reguły przekształceń. Wynik każdego przekształcenia jest pisany w wierszu niżej, numeru wiersza wyniku więc nie podano. W każdym zdaniu podkreślono litery, do których zastosowano odpowiednią regułę gramatyczną. Ponieważ zdanie S jest zdaniem końcowym i zostało ono wywiedzione z symbolu początkowego p za pomocą reguł R_1-R_5 jest więc ono zdaniem poprawnym w języku J_1.

Języki Chomsky'ego są znacznym przybliżeniem języków naturalnych oraz języków stosowanych w matematyce i dlatego mają one dość duże znaczenie praktyczne, przede wszystkim w określaniu języków maszyn matematycznych. Poświęcono im dotychczas wiele prac i monografii, zob. np. Ginsburg.

Streszczenie

Omówiliśmy języki Chomsky'ego, mające zastosowanie do formalizacji gramatyki niektórych języków naturalnych oraz języków formalnych, używanych w maszynach matematycznych.

Zadania

1. a) Podać reguły gramatyczne, pozwalające wygenerować ze słowa początkowego wszystkie formuły arytmetyczne, zawierające wszystkie nawiasy.

Wskazówka. Użyć jako symboli końcowych alfabetu następujących symboli: $(,), +, -, \cdot, /, a, b, c, \ldots, x, y, z$.

Zbadać, czy jest to gramatyka prosta?

b) Podać gramatykę generującą wszystkie formuły arytmetyczne, nie zawierające zbędnych (do jednoznacznego odczytania formuły) nawiasów zgodnie z przyjętymi regułami opuszczenia nawiasów. Uzasadnić, że jest to gramatyka prosta.

2. Podać gramatyki pozwalające ze słów początkowych wygenerować:

a) wszystkie wyrażenia poprawne rachunku zdań, w symbolice nawiasowej. Użyć jako symboli końcowych $(,) \rightarrow, \vee, \&, \equiv, p, q, r$.

b) wszystkie wyrażenia poprawne rachunku zdań zapisane w symbolice beznawiasowej.

Jako symbole końcowe przyjąć $\rightarrow, \vee, \&, \equiv, p, g, r$ (bez nawiasów).

c) Uzasadnić, że to są gramatyki proste.

§ 54. UWAGI KOŃCOWE

We wszystkich podanych przykładach języków w tym rozdziale punktem wyjściowym był skończony zbiór elementów, który był nazywany alfabetem albo słownikiem. Ciągi końcowe elementów tego zbioru były nazywane słowami lub zdaniami. Pewne słowa były wyróżnione i nazywaliśmy je albo aksjomatami, albo wyrażeniami pierwotnymi, albo też jeszcze inaczej. Z każdym językiem skojarzony był zbiór reguł nazywanych — również zależnie od przeznaczenia — produkcjami, regułami wnioskowania, czy też gramatyką. Wreszcie w zbiorze wszystkich skończonych ciągów symboli alfabetu wyróżnialiśmy pewien podzbiór wyrażeń, który można otrzymać za pomocą reguł produkcji z ustalonych wyrażeń początkowych. Podzbiór ten, zależnie od okoliczności, nazywaliśmy zbiorem twierdzeń, zbiorem zdań poprawnych, czy też inaczej.

Przykładów takich języków można oczywiście podać znacznie więcej. Nie sądzimy jednakże, aby warto je było kontynuować w nieskończoność. Wydaje się, że sytuacja w tej dziedzinie wymaga jakiegoś całościowego, systematycznego ujęcia w celu ściślejszego sprecyzowania pojęcia algorytmu. Pewne elementy takiej syntezy można znaleźć w książkach Malcewa i Davisa. Z dotychczasowych przykładów wynika, że przez algorytm rozumieliśmy opis pewnego postępowania. Postępowanie to polegało na dokonywaniu pewnych czynności na symbolach. Nic chyba nie stoi na przeszkodzie, aby czynności, o których mowa, nie ograniczać do przekształcania symboli. Wydaje się że zupełnie podobny charakter mają również wszystkie inne czynności polegające na tworzeniu z zadanych obiektów — nowych obiektów. Oczywiście struktura wszelkich czynności tego rodzaju nie jest jednolita i wydaje się, że byłoby rzeczą pożyteczną bliższe zbadanie i sklasyfikowanie tego rodzaju czynności „produkcyjnych". Wychodząc z tak określonych czynności można zastanawiać się nad językiem przydatnym do ich opisu. Być może zresztą, że tworzenie jednego uniwersalnego języka do tego celu jest niemożliwe. W każdym razie mając ściśle określoną klasę czynności oraz dokładnie określony język, służący do opisywania czynności należących do tejże klasy, można by się pokusić o ścisłe określenie pojęcia algorytmu, jako pewnych „zdań" w tym języku, opisujących w sposób jednoznaczny postępowanie na zadanych obiektach, w celu otrzymania nowych obiektów o z góry zadanych własnościach. Mając z kolei ściśle określone pojęcie algorytmu możnaby utworzyć teorię, w której pojęcie algorytmu odgrywałoby taką rolę, jak np. pojęcie zbioru w teorii zbiorów i badać różne własności algorytmów. Szczególnie interesujące wydaje się — przynajmniej z punktu widzenia zastosowań — wprowadzenie pewnych parametrów, charakteryzujących przydatność algorytmu do realizowania go określonymi środkami oraz porównywaniu algorytmów

z punktu widzenia ich realizowalności. Takim parametrem może być np. liczba operacji potrzebnych do wyprodukowania określonego napisu z napisów początkowych.

Warto zwrócić również uwagę, że w związku z zastosowaniami praktycznymi algorytmów, nieco innego znaczenia nabiera problem rozstrzygalności. Stracił on chyba w tym kontekście nieco na znaczeniu. Informacja o tym, czy dany problem jest rozstrzygalny czy nie, wobec możliwości stosowania maszyn matematycznych do realizowania algorytmów traci trochę na ważności. Jeżeli bowiem problem jest rozstrzygalny, ale rozwiązanie go za pomocą najszybszych istniejących maszyn trwałoby pięć lat, czy tysiąc lat, to z punktu widzenia zastosowań, problem taki można traktować jako „obecnie nierozstrzygalny". Z drugiej strony, jeżeli wiemy, że jakiś problem jest nierozstrzygalny, nie znaczy to wcale, że nie da się go rozwiązać. Znaczy to tylko tyle, że nie wiemy, czy zagadnienie daje się rozwiązać w z góry zadanej skończonej liczbie kroków. Natomiast jeżeli problem jest nierozstrzygalny, to niezależnie od tego ile kroków wykonaliśmy, jeżeli rozwiązanie nie zostało jeszcze znalezione nie wiadomo, czy nie znajdzie się go w dalszych krokach. Inaczej mówiąc, nie można z góry ocenić liczby kroków, po wykonaniu których można już przerwać dalsze postępowanie. Może się jednak zdarzyć, że któryś krok naszego postępowania doprowadził do rozwiązania, mimo, że problem jest nierozstrzygalny. A więc czynnikiem decydującym o tym czy jakiś problem jest „naprawdę" rozwiązalny jest nie to, czy jest on rozstrzygalny czy nie, jak to się powszechnie przyjmuje w badaniach teoretycznych, lecz to, czy w określonym okresie środkami, którymi dysponujemy udało się nam znaleźć rozwiązanie, czy też nie. A więc czynnikiem tym jest nie rozumowanie a eksperyment.

Rozdział 10

MASZYNY I ALGORYTMY

Podane w poprzednim rozdziale metody przekształcania symboli polegały na „ręcznym" wykonywaniu czynności, podanych w przepisie przekształcania. Podawane algorytmy były więc spisem instrukcji dla człowieka, który miał je wykonywać, otrzymując w rezultacie poszukiwaną odpowiedź na zadane pytanie. Uzyskanie tej odpowiedzi było sprawą czysto mechaniczną i polegało na realizowaniu zadanych wskazówek. Sprawa sensu wykonywanych czynności, jak i interpretacja uzyskanych odpowiedzi, nie należała w zasadzie do człowieka, wykonującego algorytm. Rola jego ograniczała się do czysto formalnych manipulacji na symbolach, spełniał więc on tutaj rolę maszyny, która z zadanych ciągów symboli — tworzy nowe ciągi symboli na drodze czysto formalnych przekształceń. Nic więc dziwnego, że studia nad problemami rozstrzygalności i algorytmami doprowadziły matematyków do zainteresowania się, przynajmniej teoretycznie, maszynami, które by były w stanie przekształcać formalnie zadane wyrażenia w inne wyrażenia.

Pierwszą ideę takiej maszyny, która by działała podobnie do człowieka realizującego odpowiedni przepis postępowania, podali niezależnie i mniej więcej w tym samym czasie matematyk angielski A. Turing oraz matematyk amerykański E. Post, o którym już wspominaliśmy w poprzednim rozdziale w związku z językami. Idee ich zostały opublikowane w latach 1936-7. (Patrz: A. Turing oraz E. Post [1936]).

Koncepcja Turinga i Posta stanowią podstawę dzisiejszej teorii maszyn matematycznych i automatów, jakkolwiek obaj twórcy tego kierunku, wprowadzając do matematyki pojęcie maszyny, mieli raczej na uwadze sprawy czysto matematyczne, aniżeli jakiekolwiek zastosowania praktyczne.

Nie chodziło im o podanie konstrukcji praktycznie realizowalnej maszyny, która by mogła rozwiązać jakieś rzeczywiste problemy matematyczne, a raczej uściślanie pojęcia algorytmu.

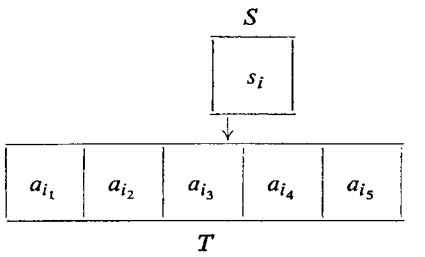
W rozdziale tym zapoznamy się z maszynami Turinga i Posta, które stanowią punkt wyjścia do określenia wielu innych maszyn studiowanych obecnie w związku z teorią maszyn matematycznych i lingwistyką matematyczną. Kilka ważniejszych przykładów takich maszyn podamy również w tym rozdziale.

§ 55. MASZYNA TURINGA

Zamiast operować pojęciem przepisu realizowanego przez człowieka, w celu przekształcenia zadanych ciągów symboli, możemy przyjąć, że proces ten odbywa się bez udziału człowieka i jest realizowany przez odpowiednie urządzenie. Znikną wtedy wątpliwości związane z tym, jakie umiejętności musi posiadać człowiek, aby był w stanie zrealizować dowolny algorytm i wiele innych problemów natury raczej psychologicznej. Pierwszy — jak już to wspominaliśmy — koncepcje takiej maszyny wprowadził Turing.

Maszynę Turinga możemy określić jako urządzenie, składające się z nieskończonej taśmy T podzielonej na kratki, jak to pokazano niżej

zwanej czasem *pamięcią maszyny*, oraz urządzenia S, zwanego *sterowaniem maszyny*. W każdej kratce taśmy T może być zapisany jeden symbol z ustalonego dla danej maszyny alfabetu $A = \{a_1, a_2, \ldots, a_n\}$; alfabet A zawiera również i symbol pusty $\varnothing$. Urządzenie sterujące S może znajdować się w jednym z k możliwych stanów $s_1, s_2, \ldots, s_k$. Zamiast mówić, że urządzenie sterujące S znajduje się w stanie s_i, będziemy mówili, że maszyna znajduje się w stanie s_i. W każdej chwili urządzenie sterujące S może „obserwować" jedną kratkę taśmy T. Symbol znajdujący się w obserwowanej kratce będziemy nazywali *symbolem obserwowanym*. Maszynę Turinga możemy sobie więc wyobrazić, jak to pokazano niżej:



Parę (s_i, a_j), gdzie s_i — aktualny stan maszyny, a_j zaś — symbol aktaulnie obserwowany przez maszynę — nazwiemy *sytuacją*. *Stany maszyny* podzielimy na *czynne* i *bierne*. Stany bierne będziemy oznaczali kreską u góry, jak np. $\bar{s}_i$. Maszyna znajdując się w stanie czynnym wykonuje ruch. Ruch powoduje zmianę

a) stanu s_{i_1} na stan s_{i_2};

b) obserwowanego symbolu a_{j_1} na a_{j_2}; w szczególnym przypadku a_{j_2} może być symbolem pustym $\varnothing$. Znaczy to, że na miejscu obserwowanego symbolu a_{j_1} zostanie wydrukowany przez urządzenie sterujące nowy symbol a_{j_2}, lub gdy a_{j_2} jest symbolem pustym symbol a_{j_1} zostanie wymazany.

c) obserwowanej kratki taśmy, na sąsiednią kratkę z lewej lub prawej strony; w szczególnym przypadku obserwowana kratka może nie ulec zmianie.

Na skutek ruchu maszyna znajduje się w nowej sytuacji (s_{i_2}, a_{j_2}).

Jeżeli maszyna znajduje się w stanie biernym — nie wykonuje ona ruchu.

Dla określenia więc działania konkretnej maszyny Turinga musimy określić wszystkie ruchy maszyny w każdej sytuacji, podając nowy stan, nowy symbol, który ma być wydrukowany przez maszynę w obserwowanej kratce oraz — nową kratkę, którą ma maszyna obserwować. Ponieważ liczba stanów, jak i symboli, jest skończona, zachowanie się maszyny możemy przedstawić w postaci tabelki, w której będą podane ruchy dla wszystkich możliwych sytuacji maszyny.

Dla przykładu rozpatrzmy prostą maszynę Turinga.

Jako alfabet maszyny przyjmiemy zbiór trzech liter $\{a, b, c\}$. Przyjmiemy, że maszyna ma pięć stanów s_0, s_1, s_2, s_3, s_4. Stany s_0-s_3 są czynne,

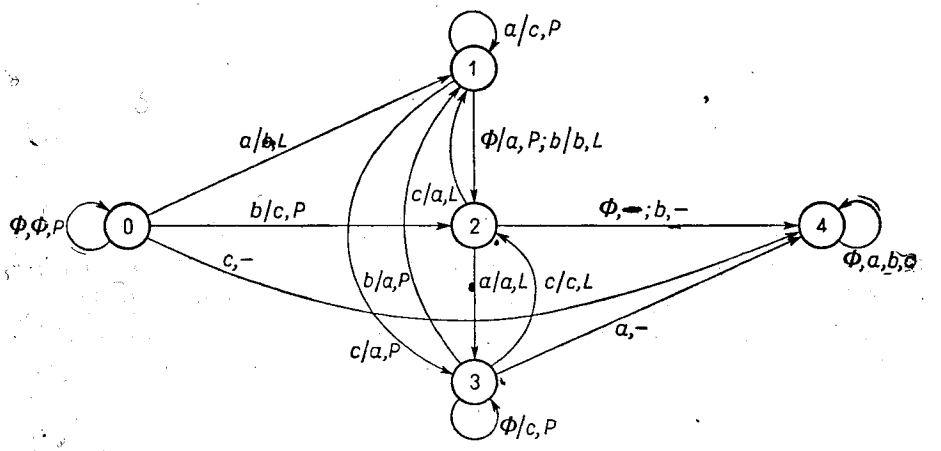
natomiast stan $\bar{s}_4$ jest stanem biernym. Dla prostoty stany maszyny będziemy oznaczali 0, 1, 2, 3, 4. Działanie maszyny jest określone tabelką:

	0	1	2	3	4
∅	0 0, P	2 *a*, P	4 —	3 *c*, P	4 —
a	1 *b*, L	1 *c*, P	3 *b*, L	4 —	4 —
b	2 *c*, P	2 *b*, L	4 —	1 *a*, P	4 —
c	4 —	3 *a*, P	1 *a*, L	2 *c*, L	4 —

W każdej kratce tabelki podano stan oraz drukowany symbol i zmianę obserwowanej kratki odpowiadającej każdej sytuacji. Litera P oznacza przejście do obserwowania prawej kratki, L — przejście do obserwowania lewej kratki, a N — niezmienienie obserwowanej kratki (ten ostatni symbol w tabelce nie został użyty). Z tabelki możemy odczytać ruch maszyny w każdej sytuacji. Na przykład jeżeli maszyna znajduje się w sytuacji 2,*a* to wykona ona ruch 3, *b*, L, co oznacza, że przejdzie ona do stanu 3, na miejsce symbolu *a* wydrukuje symbol *b* oraz przesunie taśmę o jedno miejsce w prawo, tzn. przejdzie do obserwowania pierwszej kratki z lewej strony kratki obserwowanej aktualnie. Przyjęliśmy, że w stanie biernym maszyna nie drukuje nowego symbolu oraz nie zmienia obserwowanej kratki, dlatego wszędzie w tablicy wpisaliśmy w tych przypadkach symbol „—".

Działanie maszyny Turinga wygodnie jest przedstawiać nie w postaci tablicy a za pomocą wykresu w następujący sposób: stany maszyny oznaczymy jako punkty na płaszczyźnie. Jeżeli w wyniku ruchu, maszyna przechodzi od stanu s_i do stanu s_j, to punkty odpowiadające tym stanom połączymy strzałką, skierowaną od s_i do s_j. Jednocześnie przy strzałce napiszemy, przy jakim symbolu nastąpiło przejście oraz symbol drukowany przez maszynę, a także literę P, L lub N. Wykres taki będziemy nazywali *wykresem przejść* maszyny. Dla rozpatrywanego przykładu maszyny wykres

przejść jest pokazany na rysunku 6 (na rysunku, podobnie jak w tabelce zamiast symbolu N użyto —). Z wykresu przejść można łatwo odczytać, że jeżeli np. maszyna znajduje się w stanie 0 i obserwuje symbol *a*, to w wyniku tej sytuacji zmieni ona stan na 1, wpisze na miejsce symbolu *a* symbol *b* oraz przejdzie do obserwowania lewej kratki. A więc *a*/*b*, L oznacza napisanie na miejscu symbolu *a* — symbolu *b* oraz przesunięcie taśmy

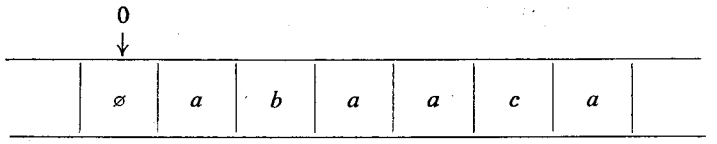


Rys. 6

o jedną kratkę w prawo. Dla uproszczenia, jeżeli maszyna przechodzi od jednego do drugiego stanu przy dwu różnych literach, na wykresie nie rysowano dwu strzałek, a tylko jedną, pisząc przy niej oba symbole, przy których to przejście następuje, jak np. ∅/*a*, P, *b*/*b*, L. To znaczy, że maszyna przechodzi ze stanu 1 do stanu 2 (patrz rysunek 6) wtedy, jeżeli odczytała symbol pusty ∅, drukując jednocześnie na jego miejscu symbol *a* i przesuwając taśmę o jedną kratkę w prawo — bądź też przejście to może nastąpić przy odczytaniu symbolu *b*; wtedy jednakże na miejsce symbolu *b* drukowany jest symbol *b* (tzn. nie ulega on zmianie) — oraz maszyna przesuwa taśmę w lewo.

Rozpatrzmy teraz w jaki sposób podana maszyna Turinga z zadanego ciągu symboli produkuje nowy ciąg symboli. Dla uproszczenia opisu, jeżeli maszyna obserwuje symbol *a* oraz znajduje się w stanie *i* — zapiszemy a_i. Na przykład a_0 oznacza, że maszyna aktualnie obserwuje symbol

a oraz znajduje się w stanie 0. Załóżmy więc, że na taśmie maszyny jest napisany ciąg *abaaca* oraz że maszyna obserwuje pierwszą pustą kratkę przed tym słowem, jak to pokazano niżej



Strzałka oznacza obserwowaną kratkę a liczba przy strzałce stan maszyny. Zgodnie z przyjętą umową fakt ten zapiszemy

$$\varnothing_0 abaaca$$

Jeżeli maszyna jest w stanie 0 i obserwuje symbol ⌀, to zgodnie z tablicą maszyny (lub wykresem przejść) — nie zmienia ona tego symbolu i przechodzi do obserwowania następnego symbolu z prawej strony, pozostając w stanie 0, co zapiszemy: $\varnothing a_0 baaca$. Pisząc kolejno w ten sposób wszystkie ruchy maszyny otrzymamy

1. $\varnothing_0 abaaca$
2. $\varnothing a_0 baaca$
3. $\varnothing_1 bbaaca$
4. $a_2 bbaaca$
5. $\varnothing_3 abbaaca$
6. $ca_3 bbaaca$
7. $ca_4 bbaaca$

Ponieważ stan 4 jest stanem biernym maszyna zatrzyma się i nie wykona już żadnego ruchu. A więc w ten sposób ze słowa początkowego *abaaca* maszyna wyprodukowała słowo końcowe *cabbaaca*.

Podany przykład maszyny nie służył do produkowania napisów mających jakiś sens matematyczny. Można oczywiście tak określić maszynę Turinga, aby realizowała ona jakieś algorytmy, dotyczące rzeczywistych teorii matematycznych, np. sprawdzała, czy zadany napis jest twierdzeniem w rachunku zdań. Tutaj chodziło nam tylko o zilustrowanie samego mechanizmu działania maszyny.

W stosunku do dowolnej maszyny Turinga możemy również pytać, jak to czyniliśmy w językach podanych w poprzednim rozdziale, czy można słowo P otrzymać ze słowa Q za pomocą określonej maszyny Turinga. Możemy również pytać, czy maszyna zatrzyma się po wykonaniu pewnej ilości operacji, jeżeli na jej taśmie napiszemy jakieś słowo w alfabecie maszyny.

Pojęcie maszyny Turinga może służyć również do określenia języka. Przyjmijmy mianowicie, że mamy zadany alfabet A oraz skończony zbiór $A' \subset A^*$. Elementy zbioru A' nazwiemy *wyrażeniami pierwotnymi* (albo *aksjomatami*). Zbiór wszystkich wyrażeń końcowych, które można wyprodukować za pomocą maszyny M z aksjomatów nazwiemy *językiem* produkowanym przez maszynę M. Język ten można interpretować jako zbiór twierdzeń jakiejś teorii, bądź zdań poprawnych w jakimś języku formalnym, czy też jeszcze inaczej.

Na maszynę Turinga można więc patrzeć z dwojakiego punktu widzenia; z jednej strony koncepcja tej maszyny powstała w związku z problemami rozstrzygalności i nadal z tą problematyką jest silnie związana — z drugiej zaś strony maszyna Turinga jest wygodnym pojęciem w formułowaniu różnych problemów dotyczących maszyn matematycznych — a przede wszystkim jest ona dobrym narzędziem do opisu czynności związanych z przekształcaniem napisów.

Istnieje hipoteza, że każdy algorytm można zrealizować za pomocą odpowiedniej maszyny Turinga. Zdanie to jest hipotezą a nie twierdzeniem z tego powodu, że pojęcie algorytmu nie jest dostatecznie sprecyzowane. W związku z powyższym równoważności między maszynami Turinga a algorytmami dowieść nie możemy, zob. rozdział XII § 67.

Streszczenie

Maszyna Turinga nie jest żadnym rzeczywistym urządzeniem technicznym, a pojęciem matematycznym, takim, jak funkcja czy liczba, wprowadzonym w związku z badaniami nad rozstrzygalnością teorii matematycznych i algorytmami. W związku z maszynami matematycznymi potęcie maszyny Turinga znalazło liczne nowe zastosowania.

Zadania

1. Rozpatrzyć działanie podanej maszyny Turinga dla następujących słów początkowych:

aba_0aca

aba_2aca

$abaac_3a$

$abaaca_2$

2. Podać tablicę i wykres przejść maszyny Turinga, produkującej z zadanego słowa, słowo, w którym symbole występują w odwrotnej kolejności do słowa oryginalnego. Ile stanów musi mieć taka maszyna?

3. Podać tablicę i wykres funkcjonalny dla maszyny Turinga, produkującej ze słowa P słowo PP.

4. Podać tablicę oraz wykres przejść dla maszyny Turinga sprawdzającej, czy w napisanym na taśmie słowie występuje zadana kombinacja symboli (np. aba).

Wskazówka. Maszyna po znalezieniu szukanej kombinacji symboli ma przejść do wyróżnionego stanu biernego $\bar{s}_i$, natomiast jeżeli poszukiwana kombinacja liter w słowie badanym nie występuje po odczytaniu ostatniego symbolu słowa, maszyna ma przejść do innego stanu biernego $\bar{s}_j$.

5. Podać maszynę Turinga, która dla dowolnej formuły nawiasowej, zawierającej wszystkie nawiasy, odszukuje funktor główny tej formuły.

Wskazówka 1. Funktor główny, to taki symbol operacji w formule, która jest wykonana ostatnia. Por. zadanie 2 z § 7 rozdział II.

Wskazówka 2. Maszyna, po znalezieniu funktora głównego ma przejść do stanu biernego.

6. Podać maszynę Turinga sprawdzającą, czy formuła zawierająca zmienne zdaniowe, symbole spójników logicznych oraz wszystkie nawiasy, jest napisana poprawnie.

Wskazówka. Jeżeli formuła jest napisana poprawnie, maszyna ma przejść do jednego stanu biernego, a po stwierdzeniu, że formuła jest niepoprawna — do innego stanu biernego.

§ 56. MASZYNA POSTA

W celu uściślenia pojęcia algorytmu, mniej więcej w tym samym czasie, niezależnie do Turinga, także Post wprowadził pojęcie maszyny (patrz Post [1936]). Maszyna Posta również składa się z nieskończonej taśmy T podzielonej na kratki. Taśma T jest obserwowana przez urządzenie sterujące S. W każdej chwili urządzenie S obserwuje tylko jedną kratkę taśmy T. W każdej kratce może być zapisany jeden z symboli ustalonego alfabetu A. Przyjmiemy, że alfabet składa się z symboli $a_0, a_1, a_2, \ldots, a_n$, gdzie a_0 jest symbolem pustym $\varnothing$ (1).

Urządzenie sterujące S maszyny Posta może wykonywać następujące czynności:

P — przejście do obserwowania sąsiedniej kratki z prawej strony.

L — przejście do obserwowania sąsiedniej kratki z lewej strony.

N — niezmienianie obserwowanej kratki (2).

Urządzenie to może również wpisywać w oznaczone kratki symbole z alfabetu A. Czynności te oznaczamy schematycznie:

$\to a_0$ ($\to \varnothing$) wpisanie do obserwowanej kratki symbolu a_0,

$\to a_1$ wpisanie do obserwowanej kratki symbolu a_1,

. . . . . .

$\to a_n$ wpisanie do obserwowanej kratki symbolu a_n.

Oprócz podanych czynności maszyna może wykonywać pewną ilość czynności, zwanych *przejściami warunkowymi*. Wyjaśnimy na czym one polegają.

Jeżeli wszystkie czynności, łącznie z przejściami warunkowymi, wykonywane przez sterowanie są jakoś ponumerowane liczbami od 1 do r, tzn.

$$C_1, \quad \ldots, \quad C_r,$$

jest ciągiem wszystkich czynności układu sterowania, to przejście warunkowe polega na tym, że jeżeli sterowanie obserwuje aktualnie symbol a_j, gdzie $j = 0, \ldots, n$, to następnie przechodzi do wykonania czynności C_{i_j}, gdzie $1 \leqslant i_j \leqslant r$. Przejście warunkowe zapisujemy symbolicznie w postaci:

(1) $$(a_0/i_0, a_1/i_1, \ldots, a_n/i_n)$$

(1) Post przyjmował, że alfabet składa się tylko z dwu symboli $\varnothing$ oraz a, jednakże mając na uwadze ewentualne praktyczne zastosowanie maszyny Posta, wygodnie jest przyjąć bogatszy alfabet.

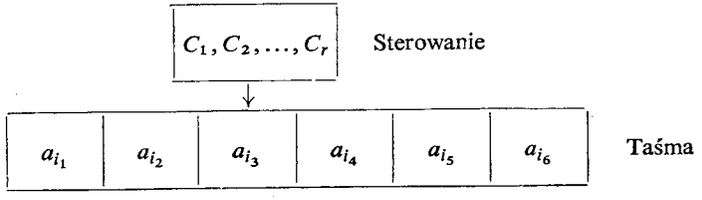
(2) Wszystkie zmiany obserwowanych kratek są podawane w stosunku do kratki, obserwowanej aktualnie.

Zapis ten mówi, że sterowanie wykonując czynność warunkową (1), jeżeli obserwuje symbol a_0, przejdzie do wykonywania czynności C_{i_0}, jeżeli obserwuje symbol a_1, przejdzie do wykonywania czynności C_{i_1} i tak dalej.

Ostatnią wreszcie czynnością jaką maszyna Posta może wykonywać jest czynność

Stop — przerwanie działania.

Maszyna Turinga była określona przez podanie dla niej tablicy przejść, natomiast maszynę Posta określimy, podając skończony ciąg czynności $C_1, C_2, \ldots, C_r$, wykonywanych przez urządzenie sterujące — taki, że każda czynność C_i jest jedną z czynności podanych poprzednio. Tak więc maszynę Posta możemy narysować następująco:



Działanie maszyny jest jednoznacznie scharakteryzowane: alfabetem, ciągiem $C_1, C_2, \ldots, C_r$ oraz aktualnie wpisanym słowem na taśmie i położeniem urządzenia sterującego na taśmie (tj. aktualnie obserwowaną kratką na taśmie przez urządzenie sterujące).

Jeżeli żadna z czynności $C_1, C_2, \ldots, C_r$, nie jest przejściem warunkowym ani czynnością — stop, to maszyna kolejno wykonuje czynności C_1, C_2, itd. aż do czynności ostatniej C_r, po czym się zatrzymuje. Jednakże jeżeli czynność C_i jest przejściem warunkowym, to po nim maszyna wykonuje ogólnie biorąc nie czynność C_{i+1}, lecz czynność, której numer zależy od aktualnie obserwowanego symbolu przez maszynę. W ten sposób maszyna może wykonywać niektóre czynności podane w ciągu $C_1, C_2, \ldots, C_r$ wielokrotnie w różnej kolejności, aż dojdzie do czynności stop. Zauważmy, że maszyna może nigdy nie dojść do jakiejś czynności, choć ta czynność występuje w ciągu $C_1, \ldots, C_r$.

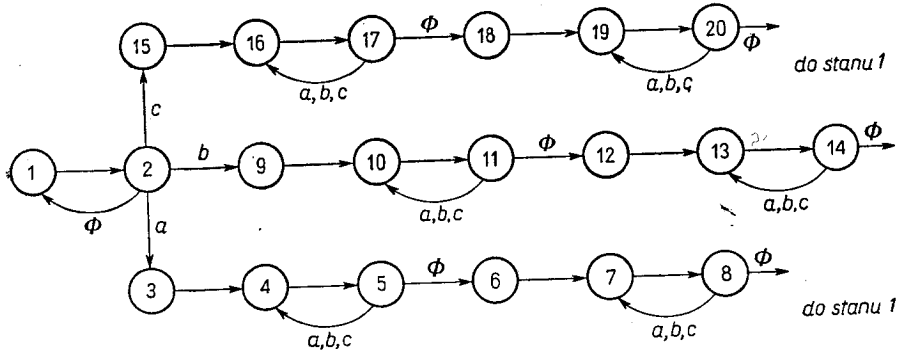
Rozpatrzmy przykład prostej maszyny Posta. Niech alfabet maszyny składa się z symboli $\varnothing, a, b, c$. Sterowanie maszyny niech będzie opisane następującym ciągiem czynności:

1. P
2. $\rightarrow(\varnothing/1, a/3, b/9, c/15)$
. . . . . .
3. $\rightarrow \varnothing$
4. P
5. $(\varnothing/6, a/4, b/4, c/4)$
6. $\rightarrow a$
7. L
8. $(\varnothing/1, a/7, b/7, c/7)$
. . . . . .
9. $\rightarrow \varnothing$
10. P
11. $(\varnothing/12, a/10, b/10, c/10)$
12. $\rightarrow b$
13. L
14. $(\varnothing/1, a/13, b/13, c/13)$
. . . . . .
15. $\rightarrow \varnothing$
16. P
17. $(\varnothing/18, a/16, b/16, c/16)$
18. $\rightarrow c$
19. L
20. $(\varnothing/1, a/19, b/19, c/19)$

Dla ułatwienia czytania, pewne grupy czynności oddzielono od siebie kropkami. Zanim jednak zajmiemy się bliżej prześledzeniem działania tej maszyny, podamy najpierw graficzny opis, przedstawiający postępowanie maszyny w różnych sytuacjach, który znacznie ułatwi nam zorientowanie się w strukturze i wzajemnych powiązaniach czynności maszyny.

Opis ten będziemy nazywali *wykresem przejść* dla maszyny Posta i będzie on nieco przypominał wykres przejść dla maszyny Turinga.

Czynności będziemy przedstawiali za pomocą punktów na płaszczyźnie, pisząc przy każdym punkcie numer odpowiadającej mu czynności. Jeżeli maszyna po czynności numer i wykonuje czynność j, to punkty i oraz j na wykresie połączymy strzałką, skierowaną od punktu i do punktu j. Jeżeli punkt na wykresie przejść przedstawia przejście warunkowe, to wychodzi



Rys. 7

z niego tyle strzałek, ile jest symboli w alfabecie maszyny. Każdą strzałką oznaczymy jedną literę alfabetu i połączymy ją z takim punktem na wykresie, który ma numer czynności odpowiadający danej literze, w przejściu warunkowym. Dla rozpatrywanej maszyny wykres przejść będzie miał postać pokazaną na rysunku 7. Dla uproszczenia zamiast rysować np. od punktu 5 do 4 trzy strzałki oznaczone symbolami a, b, c, narysowano tylko jedną strzałkę z tymi symbolami, co oznacza, że przejście od czynności 5 do czynności 4 następuje wtedy, gdy sterowanie odczyta jeden z symboli a, b lub c. Podobnie dla innych punktów. Również dla uproszczenia rysunku punkt 1 narysowano na wykresie dwukrotnie.

Teraz możemy już przystąpić do odszyfrowania działania maszyny. Załóżmy, że na taśmie zapisane jest słowo w alfabecie maszyny np. *aabaaccab* i że maszyna obserwuje pierwszy pusty symbol na taśmie, znajdujący się przed napisanym słowem. Czynność nr 1 spowoduje przesunięcie sterowania na pierwszy symbol słowa a czynność 2 spowoduje sprawdzenie przez maszynę co to jest za symbol. Zależnie od odczytanego symbolu maszyna

przejdzie do jednej z czynności 3, 9 lub 15. Ponieważ symbolem tym jest a, więc maszyna przejdzie do czynności 3, tj. wymaże ten symbol (wpisując na jego miejsce symbol pusty $\varnothing$) i przejdzie do sprawdzenia następnego symbolu na taśmie. Jeżeli jest to symbol różny od $\varnothing$, to sterowanie przesuwa się na prawo tak długo aż dojdzie do symbolu pustego $\varnothing$. Po znalezieniu symbolu pustego wpisuje ona na jego miejsce symbol a, po czym powraca na lewo aż do znalezienia pierwszego symbolu pustego z lewej strony słowa i powtarza wszystkie czynności od początku. Dla symboli b i c postępowanie maszyny jest analogiczne. Łatwo zauważyć, że tak określona maszyna realizuje szczególny przypadek tag-systemu, który ma następujący układ podstawień:

$$a \to a$$

$$b \to b$$

$$c \to c$$

oraz liczbę $k = 1$. A więc kolejne słowa na taśmie będą wyglądały następująco:

aabaccab

abaccaba

baccabaa

accabaab

ccabaaba

cabaabac

abaabacc

baabacca

aabaccab

abaccaba

. . . . .

Maszyna więc bez końca przepisuje pierwszą literę słowa na jego koniec, przy czym literę tę z początku słowa wymazuje.

W podobny sposób można określić maszynę Posta, która będzie np. wykonywała jakieś obliczenia na liczbach zapisanych na taśmie, czy też jakiekolwiek inne czynności na symbolach.

Oba sformułowania pojęcia maszyny, podane przez Turinga i Posta są równoważne w tym sensie, że opisują tę samą klasę maszyn. Różnica między nimi polega na tym, że do opisu maszyny użyte są różne pojęcia. Do jednych celów może być wygodniejszy aparat pojęciowy Posta, do innych natomiast — Turinga. Można pokazać, że każdą maszynę Posta można opisać za pomocą pojęć maszyny Turinga i odwrotnie każdą maszynę Turinga można opisać jako maszynę Posta. Dla wykazania równoważności wystarczy zauważyć, że stan maszyny Turinga może być interpretowany jako numer czynności w maszynie Posta i odwrotnie — numer czynności w maszynie Posta można uważać za stan odpowiedniej maszyny Turinga.

Streszczenie

Maszyna Posta może również służyć do „fabrykowania" z zadanych ciągów symboli innych ciągów, podobnie jak to miało miejsce w maszynie Turinga. Działanie maszyny jest określone przez podanie ponumerowanego ciągu czynności, należących do ustalonego zbioru czynności charakterystycznych dla maszyn Posta.

Zadania

1. Podać maszynę Posta realizującą następujący tag-system

$$a \to ab$$
$$b \to bc$$
$$c \to ca$$

dla $k = 2$.

2. Podać maszynę Posta, która z dowolnego słowa P w ustalonym alfabecie produkuje słowo PP.

3. Podać maszynę Posta, która tworzy sumę dwu liczb jednocyfrowych w układzie dziesiętnym.

Wskazówka. Jako alfabet przyjąć cyfry 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

4. Podać maszynę Posta, która bada, czy w zadanym słowie napisanym w alfabecie maszyny, zawarte jest ustalone słowo (np. *aac*).

5. Podać maszynę Posta, która z dowolnego słowa postaci PQR tworzy słowo PR, gdzie P i R są dowolnymi w alfabecie maszyny, Q zaś jest ustalonym słowem w tym alfabecie.

6. Maszynę, dla której zbiór podstawowych czynności jest taki sam jak dla maszyn Posta, z tą jedynie różnicą, że przejście warunkowe ma postać

$$(\varnothing/i+1,\ a/j),$$

gdzie a oznacza dowolny symbol alfabetu maszyny różny od symbolu pustego $\varnothing$, będziemy nazywali *maszyną Wanga*. Sens tej czynności jest następujący: jeżeli czynność nr i jest przejściem warunkowym i maszyna aktualnie obserwuje na taśmie symbol $\varnothing$, to po tej czynności wykonuje czynność następną, tj. czynność nr $i+1$; jeżeli natomiast maszyna obserwuje symbol a różny od symbolu pustego, to po czynności nr i wykonuje czynność j.

a) Określić maszynę Wanga realizującą tag-system

$$a \to a$$
$$b \to b$$
$$c \to c$$

dla $k = 1$.

b) Określić maszynę Wanga, która realizuje tag-system z zadania 1.

c) Określić maszynę Wanga, która realizuje to, co w zadaniu 2.

d) Określić maszynę Wanga, która realizuje to, co w zadaniu 5.

7. Możemy również określić maszynę Wanga, w której przejścia warunkowe są postaci

$$(\varnothing/j,\ a/i+1).$$

Znaczy to, że po odczytaniu symbolu pustego $\varnothing$ maszyna przechodzi do wykonania czynności nr j, a po odczytaniu symbolu a różnego od symbolu pustego maszyna przechodzi do wykonania czynności następnej o numerze $i+1$. Dla podanego przejścia warunkowego określić maszyny Wanga, realizujące punkty a), b), c), d), z zadania 6.

§ 57. MASZYNY RABINA I SCOTTA

Wychodząc z koncepcji Turinga i Posta wielu autorów podało nieco inne idee maszyn. Dwie z nich, które znalazły liczne zastosowania, przede wszystkim w maszynach matematycznych i lingwistyce matematycznej, podamy w tym i w następnym paragrafie. Źródłem obu tych koncepcji jest raczej maszyna Turinga, aniżeli maszyna Posta.

Maszyny Rabina i Scotta są pewną klasą maszyn, zwanych *automatami skończonymi* (patrz Rabin i Scott). Ponieważ istnieje wiele różnych ujęć pojęcia automatu skończonego, więc nie będziemy używali tego terminu, a dla wyraźniejszego podkreślenia, o jaką klasę automatów skończonych nam chodzi w niniejszym paragrafie, pozostaniemy przy terminie podanym w tytule paragrafu.

Maszynę Rabina i Scotta będziemy nazywali maszyną Turinga spełniającą następujące warunki:

1. Taśma maszyny jest skończona.

2. Maszyna może przesuwać taśmę tylko w jedną stronę, w prawo bądź w lewo. (Lub mówiąc inaczej urządzenie sterujące maszyny może posuwać się po taśmie tylko w jednym kierunku, w lewo bądź w prawo).

3. W zbiorze stanów S maszyny istnieje stan wyróżniony s_0, zwany *stanem początkowym* maszyny, oraz niepusty podzbiór $K \subset S$ zwany *zbiorem stanów końcowych.*

Niech A będzie alfabetem maszyny i niech A^* oznacza zbiór wszystkich słów w alfabecie A. Powiemy, że słowo $P \subset A^*$

$$P = p_0, p_1, \ldots, p_k$$

jest akceptowane przez maszynę M Rabina i Scotta, jeżeli dla maszyny M istnieje taki ciąg stanów

$$s_{i_0}, s_{i_1}, \ldots, s_{i_{k+1}}, \tag{1}$$

że $s_{i_0} = s_0$ (tzn. pierwszy ze stanów w ciągu (1) jest stanem początkowym maszyny M), oraz

$$s_{i_{j+1}} = M(s_{i_j}, p_j), \tag{2}$$

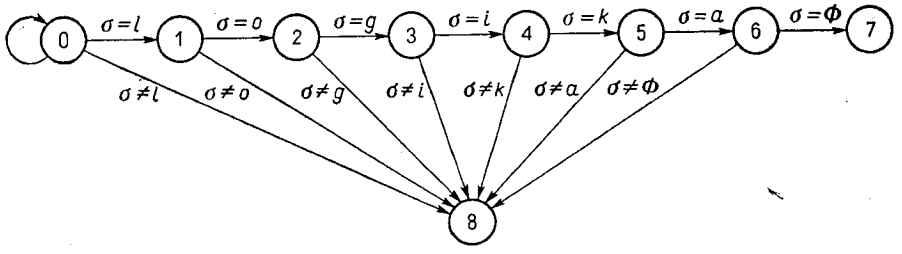
oraz $s_{i_{k+1}} \in K$.

Symbol M we wzorze (2) oznacza po prostu funkcję przyporządkowującą każdemu stanowi i odczytywanemu symbolowi nowego stanu zgodnie z tablicą maszyny. Mówiąc prościej maszyna Rabina i Scotta akceptuje słowo napisane na taśmie, jeżeli wychodząc ze stanu początkowego s_0 po przeanalizowaniu całego słowa znajdzie się w stanie końcowym. W ten sposób za pomocą maszyny Rabina i Scotta możemy określić pewien podzbiór zbioru A, zwany klasą wyrażeń akceptowanych przez maszynę M, albo też — językiem maszyny M. Maszyna ta jest bardzo wygodna do wielu zastosowań. Podamy dla przykładu maszynę Rabina i Scotta, która

spośród dowolnych słów napisanych w alfabecie łacińskim akceptuje słowo „logika". Jako alfabet maszyny przyjmiemy litery $a, b, c, \ldots, y, z$. Stany maszyny oznaczymy liczbami 0, 1, 2, 3, 4, 5, 6, 7, 8, 0 oznacza stan początkowy, 7 zaś jest stanem końcowym. Tablica maszyny będzie miała postać

	0	1	2	3	4	5	6	7	8
∅	0						7		
a						6			
b									
c									
d									
e									
f									
g			3						
h									
i				4					
j									
k					5				
l	1								
m									
n									
o		2							
p									
r									
s									
t									
u									
w									
y									
z									

W tablicy podano przyporządkowanie każdemu stanowi i literze — nowego stanu. W odróżnieniu od maszyny Turinga, maszyna Rabina i Scotta nie drukuje nowych symboli na miejscu symboli odczytanych, dlatego tablica maszyny Rabina i Scotta jest w stosunku do maszyny Turinga uproszczona. Dla wszystkich miejsc tablicy, w których nie wpisano żadnego stanu — stan ten jest stanem S. Wykres przejść dla tej maszyny jest pokazany na rysunku 8. Maszyna ta działa więc w ten sposób,



Rys. 8

że jeżeli natrafia na taśmie symbol pusty ∅ w stanie 0, to przesuwa taśmę dalej, obserwując nowy symbol. Przesuwanie taśmy następuje tak długo, aż maszyna natrafi na symbol różny od symbolu pustego, tzn. na jakąś literę alfabetu. Jeżeli jest to litera *l*, maszyna przechodzi do stanu 1, a w przypadku każdej innej litery — do stanu 8, który znaczy, że odczytane słowo nie jest słowem „logika", gdyż nie zaczyna się od litery *l*. Po literze *l* może nastąpić tylko litera *o*. Jeżeli więc w stanie 1 maszyna jako następny symbol odczyta literę *o* — przechodzi do następnego stanu 2, w przeciwnym przypadku — przechodzi do stanu 8.

Postępując podobnie dla pozostałych symboli, maszyna dochodzi do ostatniego symbolu słowa. Jeżeli jej przejścia od stanu do stanu były zgodne ze stanem 1 i 2 i ostatni stan jest stanem 7, to znaczy, że odczytała ona słowo „logika" — w przeciwnym zaś przypadku odczytane słowo było inne.

Zbiór wszystkich słów akceptowanych przez automat skończony nazywamy *językiem skończenie stanowym* (zob. Kleene [1956], Arbib, Ginsburg). W teorii automatów i języków bada się różne własności języków skończenie stanowych (patrz np. Ginsburg).

Streszczenie

Maszyna Rabina i Scotta jest maszyną Turinga, w której wprowadzono następujące ograniczenia: taśma maszyny jest skończona, maszyna może przesuwać taśmę tylko w jednym kierunku, maszyna może tylko czytać symbole na taśmie, nie może ich na taśmie natomiast drukować. Maszyna akceptuje wyrażenie napisane na tej taśmie, jeżeli poczynając od stanu początkowego po przeanalizowaniu całego napisu maszyna przejdzie do stanu końcowego. Zbiór wszystkich napisów akceptowanych przez daną maszynę Rabina i Scotta można interpretować jako język, którego gramatyka jest określona przez podanie odpowiedniej maszyny

Zadania

1. Podać maszynę Rabina i Scotta, która sprawdza, czy napisana na jej taśmie liczba jest parzysta.

2. Podać maszynę Rabina i Scotta, mającą alfabet $A = \{a, b, c\}$ oraz akceptującą wszystkie słowa postaci *abababa ... abc*, *bcbcbcbcb ... bcba*, *acacaca ... cab*.

3. Podać maszynę Rabina i Scotta o alfabecie, jak w zadaniu 2 i akceptującej wszystkie słowa typu *acPca*, *cbRbc*, gdzie P i R są dowolnymi słowami w alfabecie A, pustymi lub nie.

4. Podać maszynę Rabina i Scotta, akceptującą wszystkie słowa w alfabecie A, z wyjątkiem słów postaci *PabcQ*, gdzie P i Q są dowolnymi słowami w alfabecie A (również pustymi).

§ 58. MASZYNY WIELOTAŚMOWE

Maszyny, które przedstawimy w tym paragrafie, również pochodzą od Rabina i Scotta (patrz Rabin i Scott), jednakże dla odróżnienia od maszyn opisanych w poprzednim paragrafie, będziemy je nazywali *maszynami wielotaśmowymi*. Dla uproszczenia będziemy tu rozpatrywali pewną wąską klasę maszyn dwutaśmowych, mających jednak duże znaczenie praktyczne w teorii automatów i lingwistyce matematycznej.

Będziemy więc rozpatrywali maszyny mające dwie taśmy V oraz P. Taśmę V będziemy nazywali *taśmą wejściową* maszyny, natomiast taśmę

P — *taśmą pomocniczą*. Każda z taśm może być zarówno nieskończona jak i skończona. Z każdą z taśm jest skojarzony jeden alfabet A_V oraz A_P. Alfabet skojarzony z taśmą wejściową nazwiemy *alfabetem wejściowym* maszyny i podobnie alfabet skojarzony z taśmą pomocniczą nazwiemy *alfabetem pomocniczym*; tzn. symbole alfabetu A_V mogą być pisane tylko na taśmie V, symbole alfabetu A_P zaś mogą być pisane tylko na taśmie P. Przyjmiemy ponadto, że $A_V \subseteq A_P$, tzn. że alfabet wejściowy jest zawarty w alfabecie pomocniczym. Z każdą taśmą skojarzony jest zbiór stanów wewnętrznych maszyny, które nazwiemy odpowiednio *stanami wejściowymi* maszyny oraz *stanami pomocniczymi*. Stany wejściowe oznaczymy $v_0, v_1, \ldots, v_k$, a stany pomocnicze $p_0, p_1, \ldots, p_l$. Stany wejściowe i pomocnicze będziemy nazywali *stanami maszyny*. Taśma wejściowa może być przez maszynę tylko odczytywana. Na taśmie pomocniczej maszyna może odczytywać i zapisywać symbole. Założymy jeszcze, że taśma wejściowa jest jednostronna, tzn. maszyna może ją przesuwać tylko w jedną stronę, taśma zaś pomocnicza jest dwustronna. Jeżeli maszyna znajduje się w stanie wejściowym, to maszyna odczytuje taśmę wejściową, a jeżeli maszyna jest w stanie pomocniczym, to odczytuje albo zapisuje wtedy jakiś symbol na taśmie pomocniczej. Tak więc urządzenie sterujące maszyny może obserwować albo taśmę wejściową, albo taśmę pomocniczą, nie zaś obie taśmy jednocześnie. W zbiorze stanów wejściowych maszyny wyróżnimy stan początkowy v_0 oraz podzbiór stanów końcowych $K \subset V$, gdzie V jest *zbiorem stanów wejściowych*.

Zachowanie takiej maszyny możemy opisać identycznie jak zachowanie maszyny Turinga, bądź maszyny Rabina i Scotta, podając tablicę przyporządkowującą każdej sytuacji nowy stan oraz przesunięcie obu taśm i drukowany symbol na taśmie pomocniczej. Prościej jest jednak opisywać maszynę dwutaśmową nie za pomocą jednej tablicy, lecz przez podanie dwu oddzielnych tablic, jednej dla taśmy wejściowej, drugiej zaś dla taśmy pomocniczej (dlaczego?).

Będziemy mówili, że maszyna dwutaśmowa akceptuje słowo napisane na taśmie wejściowej

$$a_0, \quad a_1, \quad \ldots, \quad a_r,$$

jeżeli istnieje taki ciąg stanów maszyny

$$m_0, \quad m_1, \quad \ldots, \quad m_{k+1},$$

że $m_0 = v_0$, $m_{k+1} \in K$ oraz

$$m_{i+1} = M(m_i, a_i)$$

i ponadto jeżeli maszyna jest w stanie m_{k+1}, to na taśmie pomocniczej znajduje się wyróżnione słowo P zapisane w alfabecie A_P. W szczególnym przypadku P może być słowem pustym, tzn. że na taśmie pomocniczej nie jest zapisany żaden symbol alfabetu A_P. (Symbolu pustego $\varnothing$ nie będziemy zaliczali do alfabetu maszyny).

Dla ilustracji rozpatrzmy prosty przykład maszyny dwutaśmowej, mającej alfabet wejściowy $A_V = \{a, b\}$ oraz alfabet pomocniczy $A_P = \{a, b, x\}$. Działanie maszyny określone jest dwoma tablicami T_V oraz T_P.

TABLICA T_V

	v_0	v_1	v_2
$\varnothing$	v_0	v_2	v_2
a	v_1	p_1	v_2
b	p_1	p_0	v_2

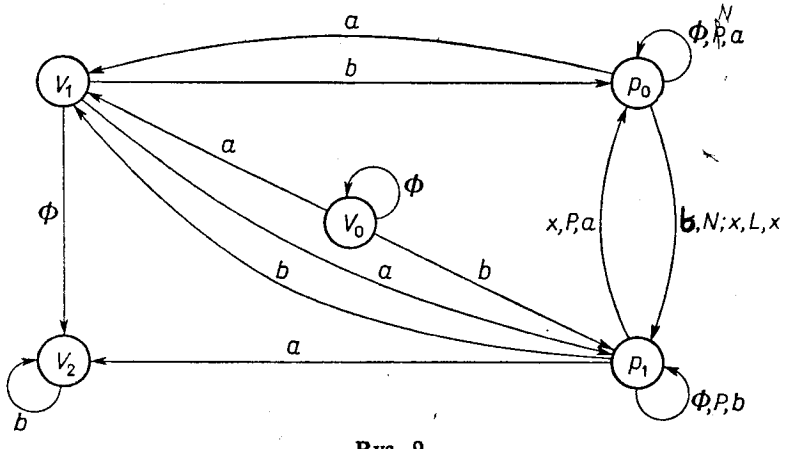
W tablicy T_V, v_0 jest stanem początkowym maszyny, zaś stan v_2 — stanem końcowym.

TABLICA T_P

	p_0	p_1	
$\varnothing$	p_0 Na	p_1 Pb	
a	v_1	v_2	
b	p_1	v_1	
b	p_1 N	v_1	
x	p_1 Lx	p_0 Pa	

W tablicy T_P, podobnie jak w maszynie Turinga, podano przy stanach pomocniczych przesunięcie maszyny do sąsiedniej kratki na taśmie (P, N, L) oraz drukowany na taśmie symbol. Na przykład jeżeli maszyna jest w stanie p_1 oraz obserwuje na taśmie pomocniczej symbol x, to drukuje ona na miejscu symbolu x symbol a i przechodzi do stanu p_0, oraz przesuwa urządzenie sterujące do kratki sąsiedniej z prawej strony. Wykres przejść dla tej maszyny jest pokazany na rysunku 9.

Rozpatrzmy jak będzie działała maszyna, jeżeli na jej taśmie wejściowej zapiszemy słowo *abaabb*. Stan maszyny będziemy pisali nad aktualnie



Rys. 9

obserwowanym symbolem alfabetu przez maszynę. Jeżeli maszyna przechodzi do obserwowania drugiej taśmy, to ostatni obserwowany symbol na pierwszej taśmie zaznaczymy kreską u góry.

	Taśma T_V	Taśma T_P
	v_0	
1.	$\varnothing abaabb$	
	v_0	
2.	$abaabb$	
	v_1	
3.	$abaabb$	
4.	$ab\bar{a}abb$	p_0
		$\varnothing$

5.		p_0
		a
	v_1	
6.	$abaabb$	$\bar{a}$
7.	$ab\bar{a}bb$	a
	v_2	
8.	$abaabb$	

Maszyna znalazła się w stanie końcowym nie przeanalizowawszy słowa wejściowego do końca. Z podanego przykładu łatwo widać sposób działania maszyny. Obserwuje ona najpierw słowo na taśmie wejściowej, symbol po symbolu, i w przypadku przejścia do stanu pomocniczego przerywa obserwowanie słowa na taśmie wejściowej — zaznaczając na niej ostatni obserwowany symbol — i przechodzi do działania na taśmie pomocniczej. Po czym w przypadku przejścia do stanu wejściowego, maszyna wraca ponownie do analizy słowa na taśmie wejściowej — od symbolu zaznaczonego — zaznaczając również ostatnią obserwowaną komórkę na taśmie pomocniczej, itd., aż do dojścia do stanu końcowego. Tak więc na przemian jest czytana to jedna, to druga taśma maszyny. Jeżeli maszyna przejdzie do stanu końcowego i na taśmie pomocniczej wyprodukuje wyróżnione słowo alfabetu pomocniczego, to słowo wejściowe jest przez maszynę akceptowane.

Można również wprowadzić tutaj pojęcie języka i badać jego własności. Istnieje wiele prac zajmujących się tym zagadnieniem. Zainteresowanego czytelnika odsyłamy do literatury (Ginsburg).

Streszczenie

Maszyna dwutaśmowa ma dwa alfabety, związane z każdą z taśm maszyny. Zbiór stanów maszyny dwutaśmowej jest podzielony na dwie klasy, w ten sposób, że każdej klasie stanów odpowiada analizowanie jednej tylko taśmy. Maszyna na przemian analizuje obie taśmy tak długo, aż znajdzie się w stanie końcowym. Na obie taśmy i sposób poruszania się maszyny po nich możemy narzucić warunki ograniczające, otrzymując w ten sposób różne klasy maszyn. W podanym przykładzie maszyna mogła

się poruszać po jednej z taśm, zwanej taśmą wejściową, tylko w jednym kierunku, odczytując na niej zapisane słowo. Na drugiej zaś taśmie, zwanej taśmą pomocniczą, dopuszczalny był ruch maszyny w obu kierunkach oraz odczytywanie jak i zapisywanie symboli. Mówiliśmy, że maszyna akceptuje słowo zapisane na taśmie wejściowej, jeżeli po przeanalizowaniu całego słowa maszyna znalazła się w stanie końcowym oraz na taśmie pomocniczej wydrukowała wyróżnione słowo w alfabecie pomocniczym.

Zadania

1. Niech słowem wyróżnionym w alfabecie pomocniczym będzie *aaa*. Podać dla maszyny dwutaśmowej, określonej w tym paragrafie, przykłady słów akceptowanych przez tę maszynę.

2. Podać maszynę dwutaśmową, akceptującą wszystkie słowa typu

1) $ababa \ldots ab$,
2) $aaa \ldots ab \ldots bbb$,
3) $abbbb \ldots bbbba$.

3. Określić maszynę dwutaśmową, akceptującą wszystkie poprawnie zbudowane formuły nawiasowe zawierające wszystkie nawiasy.

Rozdział 11

MASZYNY UNIWERSALNE

Maszyną uniwersalną dla maszyn $M_1, M_2, \ldots$ będziemy nazywali maszynę M, która może działać tak samo jak każda z maszyn M_i. Wyrażając się nieco ściślej, maszynę uniwersalną dla danej klasy maszyn możemy określić następująco: niech A_i oznacza alfabet maszyny M_i i niech A_i^* będzie zbiorem wszystkich słów w tym alfabecie. Jeżeli $P \subset A_i^*$, to przez $M_i(P)$ będziemy rozumieli słowo końcowe wyprodukowane przez maszynę M_i ze słowa początkowego P. Maszyna M jest maszyną uniwersalną dla klasy maszyny $M_1, M_2, \ldots, M_n$, jeżeli dla każdego i oraz dla każdego $P \in A_i$, $M(i, P) = M_i(P)$. Przez i, P rozumiemy tu słowo, które jest konkatenacją liczby i oraz słowa P. Musimy tu dodatkowo założyć, że łączna liczba wszystkich symboli w alfabetach maszyn $M_1, M_2, \ldots$ jest skończona.

Zwróćmy uwagę, że jeżeli np. maszyny $M_1, M_2, \ldots, M_n$ są maszynami Turinga, to maszyna M może być też maszyną Turinga, ale również może być maszyną Posta, czy też jakąkolwiek maszyną spełniającą podany wyżej warunek. Możemy więc mówić o uniwersalnej maszynie Turinga dla klasy maszyn Turinga, bądź o uniwersalnej maszynie Turinga dla klasy maszyn Posta, bądź też o uniwersalnej maszynie Posta dla klasy maszyn Turinga itp.

Ogólna zasada konstruowania maszyny uniwersalnej dla danej klasy maszyn polega na tym, że na taśmie maszyny uniwersalnej możemy umieścić opis dowolnej z maszyn M_i — oraz na takim skonstruowaniu maszyny uniwersalnej, aby dla zadanego słowa czytała ona z taśmy opis maszyny, którą ma naśladować i postępowała zgodnie z tym opisem. Opis taki można uważać za program działania maszyny uniwersalnej. Na przykład maszyna uniwersalna dla klasy maszyn Turinga będzie miała taśmę, na

której możemy zapisać np. wiersz po wierszu tablicę aktualnie rozpatrywanej maszyny Turinga i po zapisaniu dowolnego słowa z alfabetu rozpatrywanej maszyny, maszyna uniwersalna przechodzi do analizowania zapisanej tablicy i przetwarzania słowa zgodnie z podaną tablicą. Natomiast dla maszyny uniwersalnej dla klasy maszyn Posta, na taśmie maszyny uniwersalnej zapiszemy wszystkie czynności maszyny i po napisaniu na taśmie dowolnego słowa, maszyna analizuje opis umieszczony na taśmie maszyny Posta i postępuje zgodnie z tym opisem. W dalszym ciągu dla uproszczenia, maszynę uniwersalną dla klasy maszyn Posta będziemy nazywali uniwersalną maszyną Posta — niezależnie od tego czy maszyna uniwersalna jest maszyną Posta, Turinga itp., a maszynę uniwersalną dla klasy maszyn Turinga będziemy nazywali uniwersalną maszyną Turinga. Podobną terminologię przyjmiemy również dla innych maszyn.

§ 59. UNIWERSALNA MASZYNA POSTA

Jak pamiętamy z poprzedniego rozdziału, maszyna Posta mogła wykonywać następujące proste czynności:

$$\mathrm{L}, \mathrm{P}, \mathrm{N}, \rightarrow a_i, (\varnothing/i_0, a_1/i_1, \ldots, a_n/i_n), \mathrm{Stop}.$$

Napisy oznaczające te czynności będziemy nazywali *instrukcjami*. Ciąg kolejno ponumerowanych instrukcji będziemy nazywali *programem*. Przyjmiemy, że instrukcje numerujemy zawsze począwszy od zera. Przyjmiemy także, że uniwersalna maszyna Posta ma dwie nieskończone (jednostronne) taśmy. Taśmy te oznaczymy przez P oraz R. Taśmę P nazwiemy *taśmą programową*, natomiast taśmę R — *roboczą*. Alfabet, którego symbole możemy pisać na taśmie P, nazwiemy *alfabetem programu* i oznaczymy przez A_P, podobnie alfabet, związany z taśmą R, oznaczymy A_R i nazwiemy go *alfabetem roboczym*. Jako alfabet A_R możemy przyjąć dowolny ciąg symboli. Dla uproszczenia dalszych rozważań przyjmiemy, że

$$A_R = \{\varnothing, a, b, c\}.$$

$\varnothing$ tak jak poprzednio oznacza symbol pusty. Przyjęcie, że alfabet składa się tylko z czterech symboli w niczym nie ogranicza ogólności naszych rozważań. Jako alfabet A_P przyjmiemy wszystkie symbole potrzebne do zapisania dowolnej instrukcji maszyny Posta. Dla prostoty pominiemy instrukcje Stop i przyjmiemy, że maszyna zatrzyma się jeżeli wykona ostatnią instrukcję w programie.

$$A_P = \{\mathrm{N}, \mathrm{P}, \mathrm{L}, \rightarrow, \Lambda, \varnothing, a, b, c, (,), /, ', 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}.$$

Zwróćmy uwagę na pewne osobliwości tego alfabetu. Pierwsza z nich to dwa symbole puste. Pierwszy z nich Λ odnosi się do pustej kratki na taśmie P, drugi zaś $\varnothing$ oznacza pustą kratkę na taśmie R. Druga osobliwość, to występowanie przecinka jako symbolu alfabetu. Zwróćmy uwagę, że przecinki oddzielające poszczególne znaki alfabetu, do tegoż alfabetu nie należą. Trzecia wreszcie osobliwość, to włączenie do alfabetu A_P cyfr dziesiętnych, pozwalających nam na numerowanie instrukcji programu. W każdej kratce taśmy P może być zapisany tylko jeden symbol alfabetu A_P, podobnie jak to miało miejsce w maszynach dotychczas omawianych. Podobnie na taśmie R możemy zapisać w każdej kratce tylko jeden symbol alfabetu A_R.

Maszyna może analizować jedną bądź drugą taśmę, tak jak to założyliśmy omawiając maszyny wielotaśmowe (§ 58). Stany związane z taśmą P oznaczymy literami $p_0, p_1, \ldots$ itd. podobnie stany związane z taśmą R oznaczymy przez $r_0, r_1, \ldots$ itd.

Czynności związane z analizowaniem taśmy R są opisane w programie zapisanym na taśmie P. Natomiast czynności związane z analizowaniem programu są określone przez podanie tablicy maszyny. Tablicę tę, zgodnie z przyjętymi założeniami, wygodnie jest zapisać w dwu częściach, jedną dla stanów p, drugą dla stanów r.

Stan p_0 jest stanem początkowym, stany p_{11} i p_{12} stanami końcowymi. Stanu p_{12} w tablicy nie podano, nie prowadzi on bowiem do żadnego nowego stanu. Również dla przejrzystości nie wpisywano w tablicy stanu p_{11}, który powinien być wpisany we wszystkie puste kratki tablicy. Stan ten oznacza, że w programie jest błąd. Na przykład jeżeli w programie po strzałce znajdzie się przecinek, tzn. że została popełniona omyłka i maszyna przechodzi wtedy do stanu p_{11}, sygnalizując w ten sposób niewykonalność programu z powodu błędu. Również w celu uproszczenia opisu maszyny przyjęto, że po numerze instrukcji nie jest pisana kropka.

Zwróćmy uwagę na ostatni wiersz tablicy, ten, w którym stoi litera *l*. Nie ma jej w alfabecie maszyny. Wiersz ten rozumiemy następująco: jeżeli maszyna odczytała na taśmie *P* jakąś liczbę, obojętnie jaką, to przechodzi do stanów, podanych w ostatnim wierszu tablicy.

	p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}
Λ	p_0											
→		p_2										
∅			r_1		r_7							
a			r_2		r_8							
b			r_3		r_9							
c			r_4		r_{10}							
P		r_6										
L		r_5										
N		p_3										
)											p_0	
(		p_4							p_3			
/							p_9	p_7				
,				p_0					p_4		p_0	
l								p_8		p_{10}		

Dla uproszczenia również nie wpisano w tablicy przesunięcia taśmy programowej po każdym odczytanym symbolu w lewo, tzn. przejścia do obserwowania przez maszynę następnego symbolu z prawej strony. Jedynym wyjątkiem jest tutaj stan p_{10}, po którym maszyna nie przechodzi do obserwowania następnego symbolu, a odszukuje na taśmie *P* symbolu o numerze podanym w instrukcji warunkowej. Do sprawy tej jeszcze wrócimy w dalszym ciągu.

Podobnie możemy podać tablicę dla taśmy *R*.

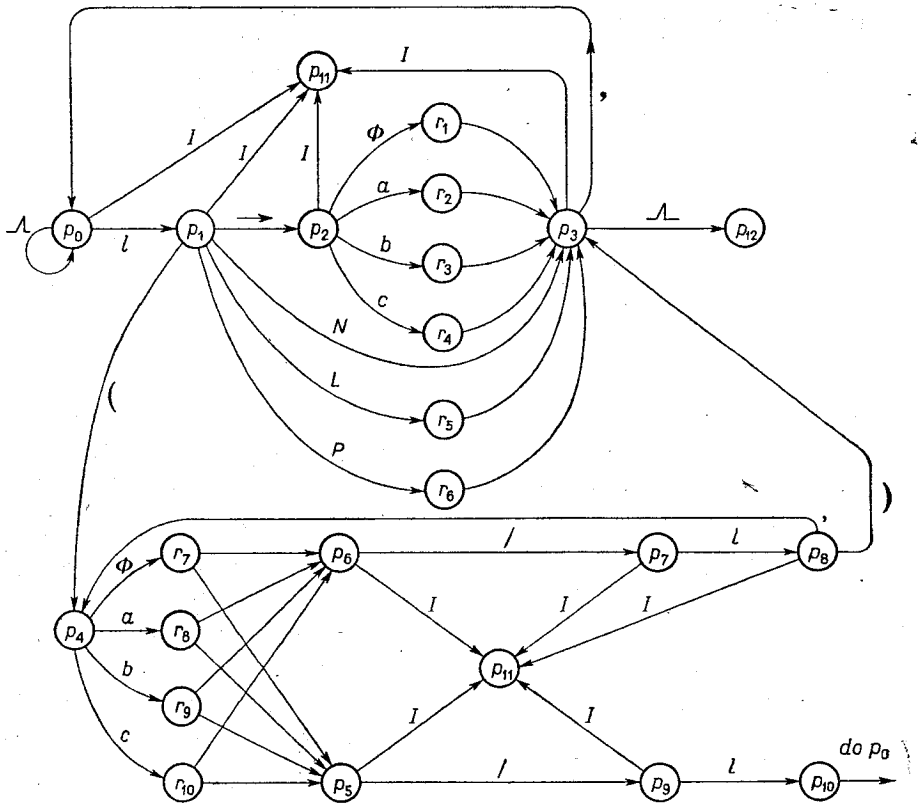
	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}
∅	$p_3, \varnothing$	p_3, a	p_3, b	p_3, c	p_3, L	p_3, P	p_5	p_6	p_6	p_6
a	—	—	—	—	—	—	p_6	p_5	p_6	p_6
b	—	—	—	—	—	—	p_6	p_6	p_5	p_6
c	—	—	—	—	—	—	p_6	p_6	p_6	p_5

Kreski w tablicy oznaczają stany, takie jak w pierwszym wierszu tablicy. Symbole po przecinku przy odpowiednim stanie maszyny oznaczają pisany na taśmie symbol bądź przesunięcie taśmy.

Zanim omówimy działanie uniwersalnej maszyny Posta, podamy jeszcze wykres przejść dla tej maszyny, co pozwoli na łatwiejsze zorientowanie się w jej strukturze. Wykres ten jest pokazany na rysunku 10. Litera *I* na rysunku oznacza symbole inne niż te, które zostały podane przy strzałkach wychodzących z danego stanu maszyny.

Maszyna zaczyna działanie od czytania taśmy *P*. Czytanie odbywa się od strony lewej do prawej. Przed rozpoczęciem czytania maszyna jest w stanie p_0. Załóżmy, że w stanie początkowym maszyna czyta symbol pusty Λ. Po odczytaniu symbolu pustego maszyna przesuwa się wzdłuż taśmy tak długo, aż natrafi na symbol różny od symbolu pustego Λ.

Jeżeli pierwszym napotkanym symbolem jest liczba, maszyna przechodzi do stanu p_1, w przypadku przeciwnym przechodzi do stanu końcowego p_{11}, sygnalizując, że program jest niepoprawny, gdyż każdy poprawny program musi się zaczynać od numeru instrukcji. Po liczbie, zależnie od instrukcji, może następować jeden z symboli →, L, P, N, (. Każdy inny symbol jest nieprawidłowy. Po strzałce może występować jeden z symboli $\varnothing, a, b, c$, tworząc w ten sposób instrukcję typu $\rightarrow a_i$. Instrukcja taka musi wywołać zapisanie odpowiedniego symbolu na taśmie *R*, co jak łatwo sprawdzić na podstawie drugiej tablicy rzeczywiście zachodzi, gdyż odczytanie jednego z symboli alfabetu A_R powoduje przejście maszyny do jednego ze stanów r_1, r_2, r_3, r_4, a każdy z tych stanów powoduje drukowanie na taśmie *R* symbolu odczytanego z taśmy.



Rys. 10.

Każda instrukcja musi być zakończona przecinkiem, po czym następuje przejście do stanu początkowego p_0 i czytanie nowej kolejnej instrukcji od początku. Gdyby po instrukcji następował symbol pusty Λ, oznaczałoby to koniec działania i przejście do stanu końcowego $\bar{p}_{12}$.

Podobnie odczytanie każdej innej instrukcji rozpoczyna się od stanu p_0 i kończy w stanie p_3. Instrukcje przesuwania L, P taśmy R w lewo i w prawo są realizowane poprzez przejście maszyny do stanów r_5 i r_6. Instrukcja warunkowa zaczyna się od nawiasu „(". Po nawiasie może wystąpić jeden z symboli alfabetu A_R, powodując przejście maszyny do jednego ze stanów r_7, r_8, r_9, r_{10}. W stanach tych, jak to widać z drugiej

tablicy, następuje porównywanie symbolu odczytanego na taśmie P z symbolem odczytanym na taśmie R. Jeżeli symbole te są jednakowe, maszyna przechodzi do stanu p_5, w przypadku zaś przeciwnym — do stanu p_6. Stany te sygnalizują więc, czy porównywane symbole są jednakowe, czy nie. Jeżeli symbole te są różne, maszyna sprawdza następny symbol programu, którym powinien być symbol „/", przechodząc w tym przypadku do stanu p_7, po czym następuje sprawdzenie, czy następnym symbolem jest liczba.

Dalszym symbolem w programie może być jedynie przecinek, bądź nawias zamykający „)". W pierwszym przypadku maszyna przechodzi do stanu p_4, powodującym dalsze analizowanie instrukcji warunkowej, w przypadku drugim instrukcja warunkowa została zakończona i maszyna przechodzi do stanu p_3, który powoduje analizowanie następnej instrukcji. Sprawdźmy teraz, co się będzie działo, jeżeli maszyna stwierdziła, że oba porównywane symbole są jednakowe. Wtedy również następuje sprawdzenie, czy następnym symbolem jest „/" i czy po nim następuje liczba. Jeżeli tak jest, maszyna przechodzi do stanu p_{10}. Stan ten powoduje odszukanie w programie instrukcji o numerze, który został ostatnio odczytany z instrukcji warunkowej i rozpoczęcie analizy programu ponownie od tej instrukcji. W jaki sposób następuje odszukanie instrukcji o zadanym numerze, nie będziemy tu bliżej wyjaśniali, gdyż wymagałoby to nieco skomplikowania maszyny i zaciemniałoby obraz całości.

Działanie maszyny uniwersalnej Posta jest bardzo proste. Na taśmie P maszyny możemy napisać jakikolwiek program i maszyna analizując ten program, symbol po symbolu program ten wykona.

Streszczenie

Maszynę, która może działać tak jak dowolna maszyna Posta nazywamy uniwersalną maszyną Posta. Maszyna uniwersalna posiada specjalną taśmę, na której możemy umieścić opis dowolnej maszyny Posta. Opis ten jest analizowany symbol po symbolu przez maszynę uniwersalną, która w ten sposób wykonuje instrukcje opisujące działanie maszyny Posta.

Zadania

1. Przeanalizować działanie uniwersalnej maszyny Posta na wykresie przejść podanym na rys. 10 w przypadku realizowania przez maszynę programu tworzącego z zadanego słowa P słowo PP.

2. Podać program dla uniwersalnej maszyny Posta, sprawdzający czy dwa słowa napisane na taśmie R są identyczne.

3. Podać program dla uniwersalnej maszyny Posta, przepisujący zadane słowo P w odwrotnym porządku.

4. Jak można określić uniwersalną maszynę Turinga?

5. Określić uniwersalną maszynę Wanga.

6. Podać do maszyny Posta instrukcję nL, nP, oznaczające przesunięcie taśmy R o n kratek w lewo bądź w prawo.
Określić taką maszynę uniwersalną.

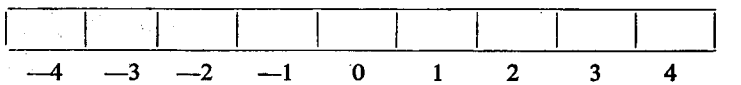
§ 60. UNIWERSALNE MASZYNY CYFROWE

Uniwersalne maszyny cyfrowe są to maszyny, które się powszechnie stosuje do wykonywania obliczeń numerycznych(1). Nie jest to więc, podobnie jak maszyna Turinga czy Posta, pojęcie matematyczne a urządzenie techniczne. Maszyn tych jednak nie będziemy tu rozpatrywali jako konkretnych urządzeń. Spróbujemy opisać ich strukturę, nawiązując do maszyn już omówionych w tej książce. Chodzi nam bowiem o to, abyśmy mogli wyrobić sobie pogląd na stosunek maszyn cyfrowych do maszyn Turinga i Posta. Dlatego też będziemy chcieli maszynę cyfrową traktować raczej jako pojęcie wygodne do opisu procesów przekształceń symboli, aniżeli jako przyrząd techniczny. W tym celu przyjęty przez nas model maszyny cyfrowej znacznie uprościmy w stosunku do maszyn rzeczywistych, starając się jednakże, aby zachować istotne własności tych maszyn z punktu widzenia, który nas w tej książce interesuje. Będziemy jednak dbać o to, aby przyjęty model maszyny cyfrowej można było komplikować, przybliżając się do rzeczywistej maszyny cyfrowej.

(1) Maszyny, o których mowa w tym paragrafie, są również maszynami uniwersalnymi dla pewnej klasy maszyn zwanych maszynami cyfrowymi. Nie będziemy tu jednak precyzować pojęcia maszyny cyfrowej a podamy od razu pojęcie uniwersalnej maszyny cyfrowej, w takiej mniej więcej postaci w jakiej przyjmuje się je powszechnie.

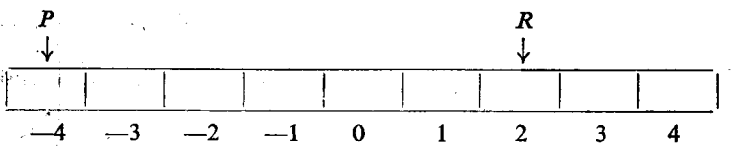
Opis maszyny cyfrowej można by zrobić podobnie do opisu uniwersalnej maszyny Posta, jednakże dla pokazania również innych możliwości określenia maszyny, podaną poprzednio metodę opisu nieco zmodyfikujemy, tak aby pozwalała ona na prostsze wyrażenie interesujących nas własności. Czytelnik zapewne zauważył, że podana w poprzednim paragrafie uniwersalna maszyna Posta jest pewnego rodzaju maszyną Turinga, do opisu działania której użyliśmy pojęcia stanu, sytuacji i innych podobnych pojęć stosowanych do opisu maszyny Turinga. Również ten sam aparat pojęciowy można by zastosować do opisu uniwersalnej maszyny cyfrowej, jednakże do tego celu jest on nieco niewygodny i dlatego musimy go nieco zmienić.

Podstawowym elementem uniwersalnej maszyny cyfrowej jest dwustronnie nieskończona taśma, podzielona na kratki. Wszystkie kratki taśmy są ponumerowane liczbami całkowitymi (ujemnymi i dodatnimi oraz zerem), jak to pokazano niżej



Numery kratek będziemy nazywali *adresami* kratek. Kratkę o adresie 0 nazwiemy *akumulatorem*. Część taśmy o adresach ujemnych nazwiemy *taśmą programów*, a część taśmy o adresach dodatnich nazwiemy *taśmą roboczą*.

Maszyna w każdej chwili obserwuje jedną z dwu kratek taśmy; jedną kratkę na taśmie programowej lub jedną kratkę na taśmie roboczej, jak to zaznaczono strzałkami niżej



Strzałki P i R mogą zmieniać swe położenie, strzałka P tylko na taśmie programowej a strzałka R na taśmie roboczej maszyny. A więc ani strzałka P ani strzałka R nie może obserwować akumulatora. Stany maszyny podzielone są na dwie klasy $p_0, p_1, \ldots, p_n$ oraz $r_0, r_1, r_2, \ldots, r_k$. Jeżeli maszyna jest w którymś ze stanów p_i, to obserwuje kratkę wskazaną przez strzałkę

P, jeżeli natomiast maszyna jest w którymś ze stanów r_i, to obserwuje kratkę wskazaną przez strzałkę R.

Jeżeli maszyna jest w stanie p_i, to może tylko odczytać symbol wskazany przez strzałkę P. Jeżeli zaś maszyna jest w stanie r_i, to może odczytać symbol zapisany w kratce wskazanej przez strzałkę R, bądź też może wpisać nowy symbol do kratki wskazanej przez strzałkę R.

Na taśmie programowej mogą być zapisywane symbole alfabetu A_P, a na taśmie roboczej — symbole alfabetu A_R. W akumulatorze mogą być zapisywane symbole obu alfabetów A_P i A_R.

Kratkę wskazaną przez strzałkę P będziemy nazywali *P-kratką*, a kratkę wskazaną przez strzałkę R *R-kratką*. Symbol zapisany w R-kratce oznaczymy σR i podobnie symbol zapisany w P-kratce oznaczymy σP.

Parę $(\sigma P, p_i)$ lub parę $(\sigma P, r_i)$ będziemy nazywali *sytuacją maszyny*. Przyjmiemy ponadto, że maszyna może nie tylko obserwować symbole, ale również badać pewne relacje między obserwowanymi symbolami. Ponieważ chcemy, aby działanie maszyny zależało nie tylko od obserwowanych symboli, ale i od zachodzących między nimi relacji, więc również parę

$$(R_i(a_{i_1}, a_{i_2}), p_j)$$

gdzie R_i oznacza pewną relację dwuargumentową, będziemy nazywali sytuacją.

Wśród stanów maszyny będziemy rozróżniali, podobnie jak poprzednio, stany czynne i stany bierne; wśród stanów czynnych wyróżnimy stan początkowy.

Jeżeli maszyna jest w stanie czynnym, wykonuje ruch. Ruch polega na zmianie:

1. położenia strzałki R;
2. symbolu obserwowanego przez strzałkę R w nowym położeniu;
3. symbolu zapisanego w akumulatorze;
4. położenia strzałki P;
5. stanu.

Wszystkie te zmiany zachodzą pojedynczo i w takiej kolejności, w jakiej zostały podane wyżej. W wyniku ruchu maszyna przechodzi do nowej sytuacji, po czym następuje nowy ruch itd., aż do przejścia do stanu biernego.

Przyjmijmy dalej jako alfabet A_R następujący zbiór symboli

$$A_R = \{\varnothing, a_1, a_2, \ldots, a_n\}.$$

Dla uproszczenia dalszych rozważań założmy, że A jest zbiorem czterech symboli $\varnothing, a, b, c$. (Jako symbole alfabetu A_R można by też przyjąć np. zbiór $\varnothing, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$, pozwalający na zapisywanie dowolnych liczb całkowitych na taśmie R). Jako alfabet A_P przyjmiemy następujący zbiór

$$A_P = \{\Lambda, \rightarrow, \mathrm{P}, \mathrm{N}, !, (,), /, ', \varnothing, a, b, c, n\}.$$

Wyjaśnienia wymaga tu symbol n. Oznacza on dowolną liczbę całkowitą dodatnią, ujemną lub zero. Zamiast tego symbolu powinniśmy właściwie podać w alfabecie A_P cyfry $0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ znak „+" i „—", a liczby całkowite traktować jako słowa złożone z cyfr i znaku. Oczywiście każda cyfra czy znak liczby powinny być zapisane w oddzielnej kratce taśmy. Jednakże dla uproszczenia przyjmiemy, że każda liczba całkowita jest przedstawiona jednym symbolem A_P i jest zapisywana w jednej kratce taśmy P lub w akumulatorze. Zwróćmy jeszcze uwagę, że gdybyśmy jako alfabet A_R przyjęli również cyfry, to alfabet A_P zawierałby już automatycznie symbole alfabetu A_R, z wyjątkiem oczywiście symbolu $\varnothing$. Znaczenie symbolu $\varnothing$ oraz Λ jest identyczne jak w maszynach omawianych poprzednio.

Przyjmiemy, że uniwersalna maszyna cyfrowa może wykonywać następujące instrukcje, zapisywane na taśmie P:

Symbol instrukcji	Nazwa instrukcji
$\rightarrow n$	odczytanie akumulatora
$\leftarrow n$	zapisanie do akumulatora
Nn	następnik
Pn	poprzednik
$!n$	przejście bezwarunkowe
$(a_0/i_0, a_1/i_1, \ldots, a_n/i_n)$	przejście warunkowe
S	Stop

Dokładne znaczenie tych instrukcji poznamy omawiając działanie maszyny. Ciąg instrukcji nazywamy *programem*. Przyjmiemy, że każde dwie instrukcje w programie, zapisane na taśmie P, są oddzielone przecinkiem.Program jest umieszczany w maszynie na taśmie P, poczynając od największych adresów ujemnych w kierunku adresów malejących. Wpisywanie programów zawsze będziemy zaczynać od adresu -1.

Program na taśmie P może mieć postać jak niżej:

	7	←	)	8	N	,	5	←		
−8	−7	−6	−5	−4	−3	−2	−1	0	1	

Pierwszą instrukcją jest $\rightarrow 5$, drugą N 8, trzecią $\rightarrow 7$. Ponieważ taśma P jest numerowana od strony prawej do lewej, więc symbole instrukcji na taśmie są pisane w odwrotnej kolejności. Wygodniej więc jest przedstawić program na taśmie w następujący sposób:

	→	5	,	N	8	,	→	7
	−1	−2	−3	−4	−5	−6	−7	−8

Dla opisania działania maszyny wprowadzimy następujące oznaczenia

P_n — ustawienie strzałki P na kratce o adresie n,

R_n — ustawienie strzałki R na kratce o adresie n,

P — przesunięcie strzałki P z kratki o adresie n do kratki o adresie $n-1$,

σP — symbol zapisany w P-kratce,

σR — symbol zapisany w R-kratce,

$\sigma 0$ — symbol zapisany w akumulatorze,

$P(\alpha)$ — zapisanie w P-kratce symbolu α,

$R(\alpha)$ — zapisanie w R-kratce symbolu α,

$0(\alpha)$ — zapisanie w akumulatorze symbolu α,

gdzie α jest dowolnym symbolem alfabetu A_P lub A_R.

Przyjmiemy, że uniwersalna maszyna cyfrowa może sprawdzać, czy symbole σP i σR są jednakowe, czy też różne, to znaczy do sytuacji maszyny oprócz $(\sigma P, p_i)$, $(\sigma R, r_i)$, należą dane sytuacje $(\sigma P = \sigma R, p_i)$ oraz $(\sigma P \neq \sigma R, p_i)$.

Zamiast opisać działanie maszyny za pomocą tablicy (tak jak maszyny Turinga czy Posta) podamy w postaci listy wszystkie interesujące nas sytuacje oraz ruchy, jakie wykona w każdej sytuacji maszyna. W pierwszej kolumnie podamy sytuację maszyny. Druga kolumna zawiera pięć rubryk. Brak symbolu w rubryce oznacza, że odpowiedni krok ruchu nie jest wykonywany i nic się nie zmienia. Symbol w rubryce oznacza, co należy wykonać w kolejnym kroku ruchu.

Lista ta pozwoli nam na uzyskanie nieco przejrzystszego opisu działania maszyny.

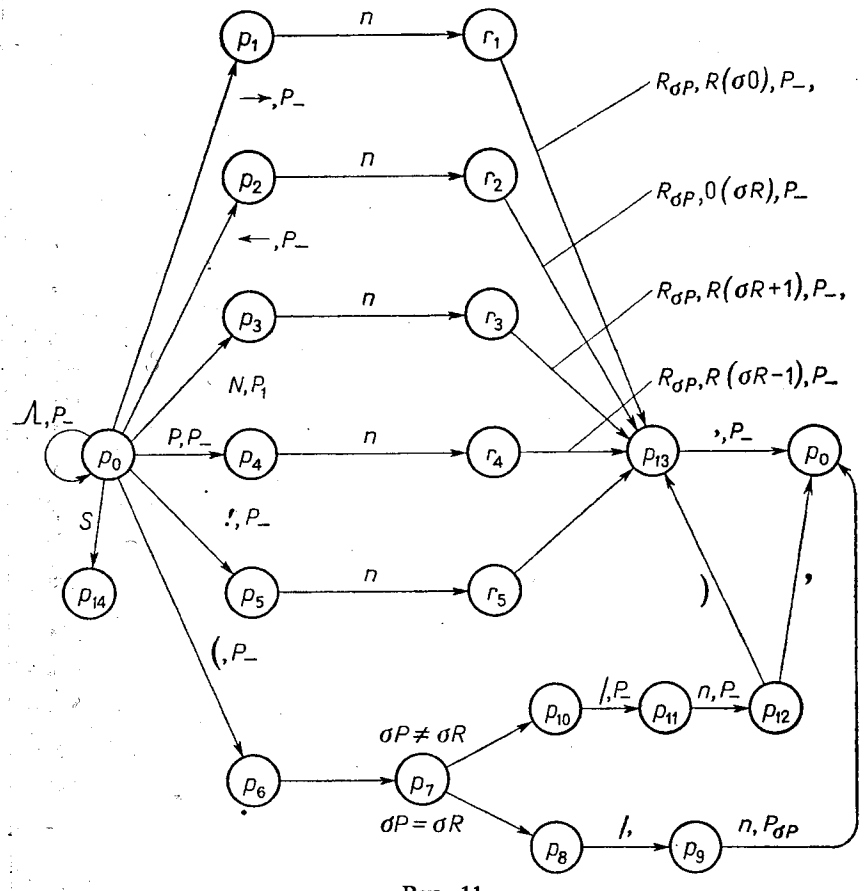
Sytuacja	Ruch maszyny				
	1	2	3	4	5
1. (Λ, p_0)				P_-	p_0
2. $(\rightarrow, p_0)$				P_-	p_1
3. (n, p_1)					r_1
4. $(-, r_1)$	$R_{\sigma P}$	$R(\sigma 0)$		P_-	p_{13}
5. $(\leftarrow, p_0)$				P_-	p_2
6. (n, p_2)					r_2
7. $(-, r)$	$R_{\sigma P}$		$0(\sigma R)$	P_-	p_{13}
8. (N, p_0)				P_-	p_3
9. (n, p_3)					r_3
10. $(-, r_3)$	$R_{\sigma P}$	$R(\sigma R+1)$		P_-	p_{13}
11. (P, p_0)				P_-	p_4
12. (n, p_4)					r_4
13. $(-, r_4)$	$R_{\sigma P}$	$R(\sigma R-1)$		P_-	p_{13}
14. $(!, p_0)$				P_-	p_5
15. (n, p_5)					r_5
16. $(-, r_5)$				$P_{\sigma P}$	p_{13}
17. $((, p_0)$				P_-	p_6

18.	(a_i, p_6)		p_7
19.	$(\sigma P = \sigma R, p_7)$	P_-	p_8
20.	$(/, p_8)$	P_-	p_9
21.	(n, p_9)	$P_{\sigma P}$	p_{13}
22.	$(\sigma P \neq \sigma R, p_7)$	P_-	p_{10}
23.	$(/, p_{10})$	P_-	p_{11}
24.	(n, p_{11})	P_-	p_{12}
25.	$(', p_{12})$	P_-	p_0
26.	$((, p_{12})$	P_-	p_{13}
27.	$(', p_{13})$	P_-	p_0
28.	(S, p)		$\overline{p}_{14}$

Wykres przejść dla tej maszyny pokazany jest na rysunku 11. Maszyna może się więc znajdować w jednej z 28 sytuacji. W każdej sytuacji wykonuje ona ruch podany w odpowiednim wierszu listy. Rozpatrzymy teraz działanie maszyny, tzn. pokażemy, w jaki sposób realizuje ona instrukcje napisane na taśmie programowej i co te instrukcje znaczą. W sytuacji nr 1 maszyna poszukuje na taśmie programowej pierwszego symbolu nie pustego. W zależności od znalezienia owego symbolu, maszyna może przejść następnie do jednej z sytuacji 2, 5, 8, 11, 14, 17, 28. Dla uproszczenia założyliśmy w tym paragrafie, że programy są napisane poprawnie i nie szukamy niedozwolonej kolejności symboli, sygnalizując je odpowiednim stanem maszyny. Rozpatrzymy każdy z tych przypadków oddzielnie, zaczynając od sytuacji 2.

Sytuacje 2, 3, 4 powodują wykonanie przez maszynę instrukcji. Łatwo sprawdzić, że instrukcja ta powoduje przepisanie symbolu zapisanego w akumulatorze do kratki o adresie n, podanym w instrukcji odczytania akumulatora.

Jak widać, z podanego opisu działania maszyny, po odczytaniu strzałki w stanie p_0, maszyna przechodzi do obserwowania następnego symbolu na taśmie programowej, przechodząc jednocześnie do stanu p_1. Po strzałce następuje liczba (adres), po odczytaniu której maszyna przechodzi do

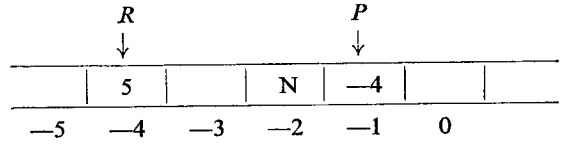


Rys. 11

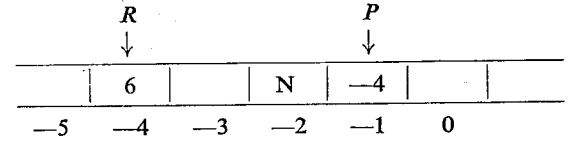
stanu r_1. W stanie r_1, niezależnie od obserwowanego symbolu na taśmie programowej (co zaznaczyliśmy pisząc w sytuacji zamiast symbolu, znak „—") maszyna wykonuje cztery czynności, $R_{\sigma P}$, $R(\sigma 0)$, P_-, p_{13}. Pierwsza z tych czynności polega na ustawieniu strzałki R na kratkę o adresie obserwowanym przez strzałkę P. Następna czynność polega na wpisaniu do kratki wskazywanej przez strzałkę R symbolu zapisanego w akumulatorze. Dalej maszyna przesuwa strzałkę P o jedną kratkę w prawo i przechodzi do stanu p_{13}. W stanie p_{13} maszyna przesuwa strzałkę P o jedno miejsce w lewo i sprawdza, czy następnym symbolem jest przecinek, a następnie przechodzi do stanu początkowego p_0.

Niemal identycznie przebiega wykonanie instrukcji $\leftarrow n$, zapisania do akumulatora. Wykonanie tej instrukcji następuje w sytuacjach 5, 6, 7. Jedyna różnica polega na tym, że obecnie maszyna przepisuje symbol z kratki wskazanej przez strzałkę R do akumulatora $(0(\sigma R))$.

Następna instrukcja Nn jest realizowana w sytuacjach 8, 9, 10. Po odczytaniu adresu maszyna ustawia strzałkę R w identyczny sposób, jak to miało miejsce w poprzednich dwu instrukcjach i do symbolu wskazywanego przez strzałkę R dodaje jedynkę. Oczywiście wykonanie tej czynności jest możliwe tylko wtedy, jeżeli symbol ten jest liczbą, a więc jeżeli strzałka R znajduje się na taśmie programowej. Instrukcja ta powoduje więc zwiększenie adresu instrukcji o jeden. Jeżeli np. strzałka P znajduje się nad liczbą 5 instrukcji $\rightarrow 5$, jak to pokazano niżej



to w rezultacie instrukcji N (—4) instrukcja $\rightarrow 5$ zostanie przez maszynę zmieniona na instrukcję $\rightarrow 6$, jak to pokazano na poniższym rysunku



Jest to bardzo ważna cecha maszyn cyfrowych. Instrukcja ta pozwala na zmianę programu napisanego na taśmie, a więc i zmianę sensu napisanych w maszynie instrukcji. Po co to jest potrzebne dowiemy się w dalszym ciągu, rozpatrując przykład programu maszyny cyfrowej.

Podobnie przebiega wykonanie przez maszynę instrukcji poprzednik, z tą różnicą, że obecnie nie dodajemy jedynki do symbolu obserwowanego przez strzałkę R, a jedynkę odejmujemy. Wykonanie tej instrukcji jest opisane sytuacjami 11, 12, 13. Na marginesie instrukcji następnik i poprzednik warto dodać, że gdybyśmy jako alfabet A_R przyjęli cyfry, to również instrukcje te mogłyby być wykonywane na symbolach zapisanych w taśmie roboczej. I jeszcze jedna uwaga. W rzeczywistych maszynach cyfrowych, operacja dodatnia jedynki i odjęcia jedynki (a także wszystkie inne operacje arytmetyczne) można wykonywać również na dowolnych symbolach a nie tylko na cyfrach. Bierze się to stąd, że w maszynie operuje się nie dowolnymi symbolami, a wszystkie symbole alfabetu maszyny są ponumerowane i maszyna zamiast symbolami alfabetu operuje ich numerami.

Ciekawa jest również następna instrukcja !n, zwana *przejściem bezwarunkowym*. Po odczytaniu tej instrukcji maszyna zmienia tylko położenie strzałki P na taśmie, ustawiając ją na kratce o adresie n, podanym w instrukcji przejścia warunkowego. Instrukcja ta jest również bardzo ważna, pozwala bowiem na wykonywanie instrukcji programu nie w takiej kolejności, w jakiej są one zapisane na taśmie programowej maszyny, a zupełnie dowolnej. Instrukcja ta jest realizowana w sytuacjach 14, 15, 16.

Najciekawszą i chyba najważniejszą ze wszystkich instrukcji jest instrukcja *przejścia warunkowego*. Instrukcja ta zaczyna się od nawiasu otwierającego. Po nim maszyna bada następny symbol a_i programu, który jest jedynym z symboli alfabetu A_R. Następnie maszyna porównuje symbol ten z symbolem wskazywanym przez strzałkę R. Jeżeli oba symbole są jednakowe, to maszyna odczytuje następujący po tym symbolu adres i ustawia strzałkę P na kratkę o odczytanym adresie i zaczyna wykonywać program od tego właśnie adresu. W przypadku przeciwnym, tj. jeżeli oba symbole, wskazywane przez strzałkę P oraz strzałkę R są różne, maszyna bada następny symbol programu. Realizowanie instrukcji warunkowej następuje w sytuacjach od 19 do 26. Na podstawie przyjętych oznaczeń nietrudno dokładnie i szczegółowo prześledzić postępowanie maszyny w tym przypadku. Sens instrukcji Stop jest oczywisty. Tak więc uniwersalna maszyna cyfrowa może wykonywać programy, składające się z ustalonych instrukcji. Nie podaliśmy tutaj instrukcji pozwalających wykonywać dowolne działania arytmetyczne, dodawanie, odejmowanie, mnożenie i dzielenie. Instrukcje nazywane arytmetycznymi będą miały postać

$$+n, \quad -n, \quad /n, \quad \cdot n.$$

Instrukcje arytmetyczne oznaczają, że należy wykonać działanie podane w instrukcji na liczbie zapisanej w akumulatorze oraz na liczbie zapisanej w kratce pod adresem n oraz wynik tego działania zapisać w akumulatorze.

Oczywiście w tym przypadku alfabet A_R musi zawierać symbole pozwalające na zapisywanie liczb na taśmie R.

Uniwersalna maszyna cyfrowa może realizować również inny zbiór instrukcji niż ten, który podaliśmy w tym paragrafie(1).

Streszczenie

Uniwersalna maszyna cyfrowa jest urządzeniem realizującym instrukcje, charakteryzujące się tym, że każda instrukcja oprócz symbolu oznaczającego, jaką operację należy wykonać na symbolach, podaje również adres miejsca na taśmie, w którym jest zapisany symbol, na którym ma być wykonana operacja.

Zadania

1. Określić maszynę cyfrową realizującą oprócz instrukcji, podanych w tym paragrafie — instrukcje arytmetyczne.

2. Określić maszynę cyfrową, której instrukcja warunkowa polega na sprawdzaniu relacji $\sigma R = \sigma 0$.

3. Określić maszynę cyfrową, która zamiast instrukcji warunkowej typu

$$(a_0/i_0, a_1/i_1, \ldots, a_n/i_n)$$

ma instrukcję warunkową postaci

$$(n/m),$$

która znaczy, że jeżeli na taśmie R w kratce o adresie n jest symbol różny od a_0, to wykonać jako następną instrukcję zaczynającą się pod adresem m, w przypadku przeciwnym przejść do następnej instrukcji.

4. Określić maszynę cyfrową realizującą instrukcję warunkową

$$p/q,$$

która oznacza, że jeżeli w akumulatorze jest symbol a_0, to przejść do instrukcji zaczynającej się pod adresem p, w przypadku przeciwnym, przejść do instrukcji zaczynającej się pod adresem q.

(1) Próby bardziej formalnego ujęcia opisu działania uniwersalnych maszyn cyfrowych znajdzie Czytelnik w pracach Elgota i Robinsona oraz Kalmára.

§ 61. PRZYKŁAD PROGRAMU MASZYNY CYFROWEJ

Rozpatrzmy teraz prosty przykład programu uniwersalnej maszyny cyfrowej(1). Załóżmy, że chcemy słowo zapisane na taśmie roboczej, począwszy od adresu n_1, przepisać na inne miejsce taśmy roboczej, tak aby pierwszy symbol słowa znajdował się po przepisaniu pod adresem n_2. Program umieścimy począwszy od adresu —1. Przyjmijmy dalej, że słowo zawiera $k+1$ symboli. Program tego zadania jest niezmiernie prosty.

Program 1

$$\leftarrow n_1,$$
$$\rightarrow n_2,$$
$$\leftarrow n_1+1,$$
$$\rightarrow n_2+1,$$
$$\ldots\ldots$$
$$\leftarrow n_1+k,$$
$$\rightarrow n_1+k.$$

Instrukcja n_1 powoduje przepisanie pierwszego symbolu słowa do akumulatora, a instrukcja następna — przepisanie tego symbolu z akumulatora do kratki o adresie n_2. Czynność ta jest powtarzana $k+1$ razy, aż wszystkie symbole zostaną przepisane na nowe miejsca. Na końcu tego programu powinna być oczywiście jeszcze instrukcja Stop. A więc przepisanie słowa k-symbolowego wymaga $2k$ instrukcji. Ściśle biorąc podany opis postępowania nie jest programem a schematem programu, gdyż w programie po strzałce „→" lub „←" muszą występować tylko liczby zapisane za pomocą cyfr ustalonego alfabetu. Mając jednakże zadane n_1, n_2 i k, możemy ze schematu tego otrzymać konkretny program.

Dla omawianego zadania możemy też podać inny program jego rozwiązania. Zauważmy, że wszystkie instrukcje tego programu są typu $\rightarrow n$ oraz $\leftarrow m$, z tą różnicą, że poszczególne instrukcje różnią się między sobą podanym w nich adresem. Adresy te w każdej parze instrukcji są zwiększane o jeden.

Dla wyjaśnienia tego programu wprowadzimy dodatkowe oznaczenia.

(1) Formalny opis pojęcia programu znajdzie czytelnik np. w pracy H. Thiele.

Niech k_i oznacza liczbę symboli, którą maszyna ma aktualnie do przepisania. Oczywiście przed rozpoczęciem działania maszyny $k_i = k+1$. Liczba k_i jest zapisana pod pewnym adresem na taśmie programowej. Oznaczmy ten adres przez $\alpha(k_i)$. Ogólnie, jeżeli p jest symbolem alfabetu A_P lub A_R zapisanym na taśmie maszyny, to adres tego symbolu będziemy oznaczali przez $\alpha(p)$. Dla ułatwienia wszystkie instrukcje ponumerujemy kolejnymi liczbami, należy jednak pamiętać, że liczby te nie są adresami.

Program 2

1. $\leftarrow n_i$,
2. $\rightarrow n_j$,
3. N $\alpha(n_i)$,
4. N $\alpha(n_j)$,
5. P $\alpha(k_i)$,
6. $(0/\alpha(\mathrm{S}), n/-1)$,
7. S.

Powyższy ciąg instrukcji nie jest również programem a schematem programu, z którego po wstawieniu wszędzie odpowiednich liczb otrzymamy konkretny program. Taki sposób pisania programów jest wygodny, pozwala bowiem tworzyć programy nie dla każdego zadania oddzielnie a dla całej grupy zadań, różniących się tylko parametrami. W ten sposób możemy podać tylko jeden schemat postępowania dla wielu zadań.

Pierwsze dwie instrukcje 1, 2 powodują przepisanie symbolu do akumulatora i z akumulatora na nowe miejsce na taśmie roboczej. Następne dwie instrukcje 3 i 4 zwiększają adresy instrukcji 1 i 2 o jeden. Instrukcja 5 zmniejsza liczbę k_i o 1, wskazując liczbę przepisywań, która pozostała jeszcze do wykonania. Instrukcja 6 sprawdza, czy wszystkie symbole zostały już przepisane. Jeżeli tak, to maszyna przechodzi do instrukcji 7 — Stop, jeżeli nie, to program jest wykonywany od początku. Przed rozpoczęciem wykonywania programu $n_i = n_1$, $n_j = n_2$, $k_i = k$.

Streszczenie

Dzięki instrukcji warunkowej maszyna może automatycznie zmieniać znaczenie instrukcji napisanych na taśmie programowej co pozwala na

znaczne skrócenie programów i stanowi istotę nowoczesnych maszyn cyfrowych.

Zadania

1. Napisać program 1 dla $n_1 = 1$, $n_2 = 10$ oraz $k = 5$. Prześledzić dla tego programu dokładnie wszystkie ruchy maszyny.

2. Napisać dla tych samych parametrów program 2.

3. Napisać program przepisujący słowo Q na taśmie R w odwrotnym porządku.

4. Napisać program sprawdzający, czy w słowie Q występuje zadany symbol alfabetu.

5. Napisać program sprawdzający, czy dwa słowa Q i R są jednakowe.

6. Napisać program wpisujący w słowie P na miejsce każdego symbolu a, słowo Q. (Symbol a występuje w słowie P)

§ 62. UWAGI KOŃCOWE

Podany przykład maszyny cyfrowej jest daleko idącym uproszczeniem rzeczywistych maszyn cyfrowych, tym niemniej, jak sądzimy, pozwala ona na zorientowanie się w zasadzie działania tych maszyn. Rzeczywiste maszyny mogą wykonać znacznie większą liczbę różnych instrukcji niż te, które podaliśmy w naszym przykładzie. Ogólna zasada ich działania pozostaje jednak bez zmiany.

Próbowaliśmy opisać maszynę cyfrową w sposób podobny, jak opisywaliśmy działanie maszyny Posta i Turinga, oraz uniwersalnej maszyny Posta. Pozwala to, jak sądzimy, na zauważenie pewnych różnic i analogii między omawianymi tu maszynami.

Streszczenie

Pojęcie maszyny cyfrowej można traktować jako pewne nowe pojęcie algorytmu wygodne do praktycznego opisywania różnych manipulacji na symbolach.

Rozdział 12

FUNKCJE REKURENCYJNE

Rozdział ten poświęcony jest trzeciej definicji pojęcia algorytmu opartej o metodę arytmetyzacji. W paragrafie 63 podana jest aksjomatyka arytmetyki, w paragrafie 64 — określenie funkcji obliczalnych, w paragrafie ostatnim 67 — uzasadnienie, że funkcje obliczalne są to właśnie te funkcje, dla których mamy efektywną metodę obliczania wartości funkcji dla danego argumentu. Termin „funkcje rekurencyjne" historycznie najwcześniejszy użyty jest tylko w tytule rozdziału, jako ogólnie znany. W tekście używamy precyzyjniejszego i nowszego terminu: „funkcje obliczalne", zgodnie z naszą literaturą zob. Grzegorczyk [1961] Mostowski. (Odpowiednikiem w literaturze angielskiej jest termin — general recursive functions, w rosyjskiej — rekursivnyje funkcje). W tekście podane są tylko najelementarniejsze fakty dotyczące funkcji obliczalnych. Część materiału znajduje się w zadaniach. Pojęcie zbioru obliczalnego, rekurencyjnie przeliczalnego podane jest w § 65. W zadaniach do § 64 naszkicowane są kwestie związku między pojęciem obliczalności określonym przez funkcje obliczalne i pojęciem obliczalności funkcji, określonym za pomocą maszyny Turinga. Precyzyjne sformułowanie i dowody związane z kwestiami arytmetycznego pojęcia obliczalności i pojęcie obliczalności za pomocą maszyn Turinga znajdzie czytelnik w książce Kleene'a [1952] jak również Malcewa, Davisa.

W paragrafie 66 znajduje się szkic arytmetyzacji teorii polegającej na przypisaniu każdemu wyrażeniu poprawnemu teorii, numeru (liczby naturalnej) i wypowiadaniu twierdzeń o jakiejś klasie wyrażeń jako twierdzeń arytmetyki o zbiorze numerów przypisanych wyrażeniom tej klasy. W oparciu o metodę arytmetyzacji podana jest ścisła definicja co to znaczy, że istnieje metoda rozstrzygania czy wyrażenie jest twierdzeniem teorii

czy też nie. Definicja ta uzyskana jest w oparciu o pojęcie obliczalności.

Funkcje obliczalne stanowią ważny dział logiki i matematyki pozwalający na precyzyjne formułowanie wielu zagadnień dotyczących algorytmów. Jedno ze ścisłych sformułowań, co to znaczy, że istnieje jakaś metoda efektywnego uzyskiwania wyniku, oparte jest o funkcje obliczalne. Istnieje hipoteza, że jest to najszersze z możliwych pojęcie efektywności (zob. par. 67). Obszerne wiadomości o funkcjach obliczalnych znajdzie czytelnik w książce Grzegorczyka [1961], Kleene'a [1952] Malcewa, Davisa. Nieco inny charakter ma książka Péter.

§ 63. LICZBY NATURALNE

W paragrafie tym podamy aksjomatyczne określenie liczb naturalnych. Podane poniżej aksjomaty zwane aksjomatami Peano opierają się na trzech pojęciach pierwotnych: zbiorze N liczb naturalnych, liczbie naturalnej 0 i funkcji $S(n)$ określonej na liczbach naturalnych o wartościach naturalnych, zwanej następnikiem. Funkcję tę interpretujemy jako funkcję $S(n) = n+1$ przyporządkowującą każdej liczbie naturalnej liczbę o jeden większą.

A oto aksjomaty Peano:

N_1. Zero jest liczbą naturalną.

N_2. Każda liczba naturalna n ma następnik $S(n)$, który również jest liczbą naturalną.

N_3. Zero nie jest następnikiem żadnej liczby naturalnej n.

N_4. Jeżeli następniki $S(n)$ i $S(m)$ dwóch liczb naturalnych n i m są równe, to i liczby n i m są równe.

N_5. Każdy zbiór X liczb naturalnych zawierający liczbę 0 i mający tę własność, że wraz z dowolną liczbą naturalną n zawiera jej następnik $S(n)$, jest zbiorem wszystkich liczb naturalnych.

Ostatni aksjomat jest szczególnie ważny i nazywa się *zasadą indukcji.*

Poniżej podajemy jeszcze te same aksjomaty zapisane w postaci sformalizowanej.

N_1. $0 \in N$,

N_2. $\mathrm{A}n\,[(n \in N) \Rightarrow (S(n) \in N)]\ \&\ \mathrm{A}n\,\mathrm{A}m\,[(n \in N)\ \&\ (m \in N)\ \&\ (m = n) \Rightarrow S(n) = S(m)]$,

N_3. $An\,[(n\in N)\Rightarrow (S(n)\neq 0)]$,

N_4. $Am\,An\,[(n\in N)\,\&\,(m\in N)\Rightarrow (S(m)=S(n)\Rightarrow (m=n))]$.

Aksjomaty N_1—N_4 są formułami języka teorii elementarnych. Aksjomat N_2 mówi, że następnik jest funkcją określoną na zbiorze liczb naturalnych o wartościach ze zbioru liczb naturalnych.

Aksjomat N_3 mówi, że 0 nie jest wartością tej funkcji. Aksjomat N_4 mówi, że funkcja ta jest różnowartościowa, tzn. że różne liczby naturalne mają różne następniki.

Aksjomat N_5 — zasada indukcji, nie jest formułą języka elementarnego. W jego wysłowieniu jak widzimy poniżej, występuje kwantyfikator AX (pierwszy kwantyfikator obejmujący zasięgiem całe wyrażenie), gdzie zmienna X przebiega już nie elementy ale zbiory elementów.

$$N_5.\quad AX\big[(n\in X)\,\&\,(0\in N)\,\&\,Am\,[(m\in X)\Rightarrow (S(m)\in X)]\Rightarrow$$
$$\Rightarrow An\,[(n\in N)\Rightarrow (n\in X)]\big].$$

Pojęcia pierwotne użyte w aksjomatach pozwalają zdefiniować dla liczb naturalnych działanie „+" i „·" (dodawania i mnożenia) oraz relację „$\leqslant$" (mniejsze, równe). Definicje działań są indukcyjne.

Korzystając z zasady indukcji możemy się przekonać, że w ciągu 0, $S(0)$, $S\big(S(0)\big)$, $S\big(S\big(S(0)\big)\big)$, $S\big(S\big(S\big(S(0)\big)\big)\big)$, ... występują wszystkie liczby naturalne i każda tylko jeden raz. Liczby te piszemy zazwyczaj za pomocą symboli

$$0, 1, 2, 3, 4, \ldots$$

oznaczając liczbę $S(0)$ symbolem 1, liczbę $S\big(S(0)\big)$ równą $S(1)$ symbolem 2 i tak dalej.

Funkcja $f(n, m)$ taką, że dla $n, m\in N, f(n, m)\in N$, spełniająca warunki:

$$(1)\qquad An[f(n, 0)=n]$$

$$(2)\qquad An\,Am[f\big(n, S(m)\big)=S\big(f(n, m)\big)]$$

określa działanie zwane dodawaniem. Zamiast $f(n, m)$ piszemy $n+m$.

Można udowodnić (również korzystając z zasady indukcji) że istnieje dokładnie jedna funkcja spełniająca zadane warunki (1) i (2). Twierdzenie to nosi nazwę *twierdzenia o istnieniu i jednoznaczności dodawania.*

Z definicji dodawania dowodzi się powszechnie znanych praw:

Prawa ekstensjonalności:

Jeżeli $n=n'$ i $m=m'$, to $n+m=n'+m'$ dla dowolnych $n, m, n', m'\in N$;

przemienności dodawania:

$$n+m=m+n \text{ dla dowolnych } n, m\in N;$$

łączności dodawania:

$$(n+m)+k=n+(m+k) \text{ dla dowolnych } n, m, k\in N;$$

oraz prawa mówiącego, że *zero jest elementem neutralnym* (zerem) *dla dodawania*:

$$n+0=n \text{ dla każdego } n\in N.$$

jak również *prawa skracania dla dodawania*:

Jeżeli $n+m=k+m$, to $n=k$, dla dowolnych $n, m, k\in N$.

Za pomocą dodawania nietrudno już określić mnożenie. Mnożenie jest to taka funkcja $g(n, m)$ o argumentach $n, m\in N$ i wartości należącej do N, która spełnia warunki:

$$(3)\qquad g(n, 0)=0,$$

$$(4)\qquad g\big(n, S(m)\big)=g(n, m)+n.$$

Z aksjomatów arytmetyki dowodzi się istnienia i jednoznaczności mnożenia. Pisze się zazwyczaj $n\cdot m$ zamiast $g(n, m)$. Dowodzi się znanych praw arytmetycznych dla tej funkcji:

prawa ekstensjonalności:

Jeżeli $n=n'$ i $m=m'$, to $n\cdot m=n'\cdot m'$ dla dowolnych $n, m, n', m'\in N$;

prawa przemienności:

$$n\cdot m=m\cdot n \text{ dla każdego } n, m\in N;$$

prawa łączności

$$(n\cdot m)\cdot k=n\cdot(m\cdot k) \text{ dla każdego } n, m, k\in N;$$

prawa rozdzielności mnożenia względem dodawania:

$$n\cdot(m+k)=(n\cdot m)+(n\cdot k) \text{ dla dowolnych } n, m, k\in N;$$

prawa mówiącego, że $S(0)$ (równe 1) *jest elementem naturalnym* (jednością) *dla mnożenia:*

$$n \cdot S(0) = n \text{ dla każdego } n \in N.$$

Jeżeli $n \neq 0$, to jeżeli $n \cdot m = n \cdot k$, to $m = k$ dla dowolnych $n, m, k \in N$.

Z podanych praw można wywnioskować również, że

$$n \cdot 0 = 0 \text{ dla każdego } n \in N.$$

Zazwyczaj liczby naturalne zapisujemy w notacji dziesiętnej. Znany i bardzo łatwy przepis określa operację pozwalającą z zapisu liczby n, cyfra po cyfrze, poczynając od ostatniej najmniej znaczącej, otrzymać zapis liczby $S(n)$ (następnika liczby n). Znany przepis pozwalający z zapisów dziesiętnych liczb n i m znaleźć zapis liczb $n+m$ daje się sprowadzić dzięki indukcyjnej definicji dodawania, do wielokrotnego stosowania przepisu na znajdowanie następnika.

Definicja dodawania sama w sobie stanowi już taki przepis. Podobnie znajdowanie postaci dziesiętnej iloczynu daje się sprowadzić do wielokrotnego znajdowania sumy (a więc i do wielokrotnego znajdowania następnika) dwóch liczb podanych w zapisie dziesiętnym.

Streszczenie

Podaliśmy aksjomatykę Peano liczb naturalnych, określiliśmy dodawanie i mnożenie oraz podaliśmy prawa spełnione przez te działania. Istnieją bardzo proste przepisy wykonywania tych działań na zapisach dziesiętnych liczb naturalnych.

Zadania

1. Określmy funkcję $h(m, n)$ o argumentach n, m, naturalnych i wartości naturalnej, w następujący sposób:

$$h(n, 0) = S(0) \qquad h(n, S(m)) = h(n, m) \cdot m.$$

Co to za funkcja?

Udowodnić, że

$$h(n, m+k) = h(n, m) \cdot h(n, k).$$

Co to za funkcja?

2. Określmy relację $n \mathrm{R} m$, dla $n, m \in N$ w następujący sposób:

$$(n \mathrm{R} m) \equiv \mathrm{Er}\,(r \in N \,\&\, (n+r = m)).$$

Udowodnić, że relacja ta jest zwrotna, słabo antysymetryczna i przechodnia. Udowodnić, że

$$(n \mathrm{R} m) \Rightarrow (n+k) \mathrm{R} (m+k),$$

$$(n \mathrm{R} m) \Rightarrow (n \cdot k) \mathrm{R} (m \cdot k).$$

3. a) Udowodnić, że dla każdego n: $S(n) = n + S(0)$.

b) Udowodnić, że $S(n+m) = S(n)+m = n+S(m)$, dla każdych $n, m \in N$.

c) Korzystając z b) i definicji dodawania udowodnić, że dla każdych n i m: $n+m = m+n$ (prawo przemienności dodawania liczb naturalnych).

d) Oznaczmy $S(0) = 1$, $S(1) = 2$, $S(2) = 3$, $S(3) = 4$.

Udowodnić, że:

$$2+2 = 4.$$

Wskazówka. Zapisać sumę 2+2 w postaci:

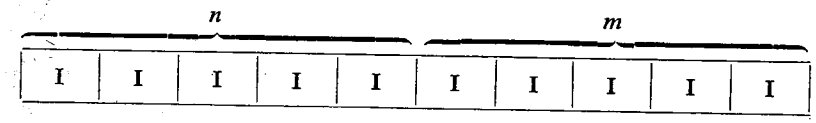
$$S(S(0))+S(S(0))$$

i skorzystać z punktu b).

4. Przyjmijmy następujący zapis dla liczb naturalnych. Zapisujemy 0 w postaci | (jednej pałeczki). Funkcja $S(n)$ polega na dopisaniu na końcu ciągu pałeczek jeszcze jednej pałeczki.

a) Podać tabelkę maszyny Turinga (zob. rozdział X, § 55) realizującej funkcję $S(n)$, gdy na taśmie wypiszemy ciąg n pałeczek.

b) Podać tabelkę maszyny Turinga realizującej z zapisu liczb n i m na taśmie w postaci



zapis liczb $n \cdot m$.

c) Podać tabelkę maszyny Turinga realizującej na taśmie zapis największego wspólnego dzielnika liczby n i m.

Wskazówka. Oprzeć się na algorytmie Euklidesa.

5. a) Podać tabelkę maszyny Turinga realizującej na taśmie zapis następnika $S(n)$ liczby naturalnej n zapisanej w zapisie dwójkowym (w postaci ciągu zer i jedynek).

b) Podać tabelkę maszyny Turinga realizującej na taśmie sumy dwóch liczb zapisanych w zapisie dwójkowym tak, że między nimi znajduje się jedna kratka wolna.

c) To samo podać dla mnożenia.

6. a) Wychodząc z zapisu dziesiętnego liczby n, opisać przepis pozwalający otrzymać zapis dziesiętny następnika $S(n)$.

b) Podać tabelkę maszyny Turinga realizującej następnik liczby naturalnej zapisanej w układzie dziesiętnym.

7. a) Udowodnić, że aksjomat N_5 — zasada indukcji — jest równoważny następującemu twierdzeniu zwanemu *zasadą minimum*:

Każdy zbiór niepusty liczb naturalnych zawiera element najmniejszy.

b) Udowodnić, że następujące twierdzenie jest równoważne zasadzie indukcji:

Jeżeli podzbiór X zbioru N ma następujące własności $0 \in X$ i dla każdego n z tego, że dla każdego k zachodzi implikacja $(k < n \Rightarrow k \in X)$ wynika, że $n \in X$, to X jest zbiorem wszystkich liczb naturalnych.

c) Zapisać zasadę minimum oraz zasadę z punktu b) tego zadania w postaci sformalizowanej za pomocą kwantyfikatorów spójników logicznych i symboli „=", „$\in$", „<".

Uwaga. Relacja $n \leqslant m$ jest relacją R z zadania 2. Relację $n < m$ definiujemy jako $n \leqslant m \,\&\, n \neq m$.

§ 64. FUNKCJE OBLICZALNE

W paragrafie tym podamy definicję takiej klasy funkcji określonych na zbiorze liczb naturalnych, przyjmujących wartości z tegoż zbioru, że dla każdej funkcji tej klasy potrafimy podać metodę obliczania wartości funkcji. Oczywiście sama definicja tej klasy będzie określać co to znaczy, że dla dowolnej funkcji istnieje metoda obliczania jej wartości w dowolnym punkcie.

Definicję poprzedzimy szeregiem uwag.

1. Przypuśćmy, że mamy takie dwie funkcje $f(x, y, z, u)$ i $g(x, y, u)$ o wartościach i argumentach będących liczbami naturalnymi, że dla każdego układu argumentów mamy podaną metodę obliczania wartości funkcji. Wtedy dla złożenia tych funkcji:

$$n(x, y, u) = f\big(x, y, g(x, y, u), u\big),$$

łatwo podać metodę obliczania wartości funkcji.

2. Oczywiście składanie funkcji nie jest jedyną możliwą operacją na funkcjach. Mając dwie funkcje $f(x)$ i $g(x, y, z)$ możemy określić nową funkcję $h(x, y)$ w sposób następujący:

$$h(0, x) = f(x),$$

$$h(y+1, x) = g\big(x, y, h(x, y)\big).$$

Poniższy sposób określania funkcji nazywa się *schematem rekurencyjnym* (dokładniej, według powszechnie przyjętej terminologii *schematem rekursji prostej*, patrz Grzegorczyk [1]). Jasne jest, że mając metody obliczania wartości funkcji $f(x)$ i $g(x, y, z)$ w dowolnym punkcie, łatwo podać metodę obliczania wartości funkcji $h(y, x)$ w dowolnym punkcie. Przypominamy, że wszystkie rozważane funkcje są funkcjami, których argumenty i wartości są liczbami naturalnymi.

3. Inny możliwy sposób tworzenia nowych funkcji z danych to przyjście od postaci uwikłanej do jawnej. Mając funkcję $f(x, y)$ szukamy funkcji $h(x)$ takiej, że $f\big(x, h(x)\big) = 0$.

Oczywiście, by funkcja $h(x)$ istniała musi być spełniony warunek:

(1) dla każdego $x \in N$, istnieje $y \in N$, takie, że $f(x, y) = 0$.

Przy spełnieniu tego warunku funkcja $h(x)$ taka, że $f\big(x, h(x)\big) = 0$, istnieje, nie musi być jednak określona jednoznacznie. Jeżeli dla jakiegoś x zarówno $F(x, y_1) = 0$ jak i $F(x, y_2) = 0$ oraz $y_1 \neq y_2$, to można równie dobrze przyjąć $h(x) = y_1$ jak i $h(x) = y_2$.

Trzeba określić jakąś operację pozwalającą w przypadku gdy spełniony jest warunek (1) jednocześnie wyznaczyć funkcję $h(x)$, taką że $f\big(x, h(x)\big) = 0$. Aby uzyskać jednoznacznie określoną wartość funkcji $h(x)$, najlepiej posłużyć się operacją

(2) $h(x) =$ najmniejsze $y \in N$ takie, że $f(x, y) = 0$.

Jeżeli funkcja f spełnia warunek (1), to tak określona operacja nazywa się *operacją minimum efektywnego* dla funkcji f.

Widoczne jest, że jeżeli mamy metodę obliczania wartości funkcji $f(x, y)$ dla każdej pary x i $y \in N$, to mamy również metodę obliczania wartości funkcji $h(x)$, powstałej przez operację minimum efektywnego, dla każdego $x \in N$.

Metoda ta polega na obliczaniu dla danego x kolejno wartości $f(x, 0)$, $f(x, 1)$, $f(x, 2)$, ... tak długo, aż natrafimy na pierwsze z kolei n takie,

że $f(x, n) = 0$. Warunek (1) zapewnia, że po skończonej, aczkolwiek nie określonej z góry, liczbie kroków takie n znajdziemy. To znalezione n będzie wartością funkcji $h(x)$ w rozważanym przez nas punkcie x.

Dla funkcji następnika $S(x) = x+1$, istnieje metoda obliczania wartości funkcji dla każdego argumentu (por. koniec paragrafu 63. Oczywiście dla funkcji tożsamościowych $I_1(x) = x$, $I_2(x, y) = y$ oraz funkcji stałej $0(x) = 0$, również istnieje metoda pozwalająca obliczyć wartość funkcji dla każdego argumentu.

Przyjmuje się następującą definicję:

DEFINICJA 1. *Funkcjami obliczalnymi* nazywamy funkcje

$$S(x) = x+1, \quad I_1(x) = x, \quad I_2(x, y) = y, \quad 0(x) = 0 \tag{3}$$

oraz wszystkie funkcje, które da się otrzymać z funkcji wymienionych w punkcie (3) za pomocą skończonej ilości operacji:

1. składania,
2. rekursji prostej,
3. minimum efektywnego.

Inaczej mówiąc, klasa funkcji obliczalnych jest to najmniejsza klasa zawierającą funkcje (3) i zamknięta ze względu na operacje wymienione w punktach 1, 2, 3. Z podanych przez nas poprzednio rozważań wynika, że dla funkcji obliczalnych mamy metody obliczania wartości funkcji w skończonej liczbie kroków.

Podamy teraz kilka przykładów funkcji obliczalnych. Z rozważań paragrafu 1 wynika, że funkcje $f(x, y) = x+y$ oraz $g(x, y) = x \cdot y$ są obliczalne. Obliczalne są również wszystkie funkcje stałe $f(x, y, z, \ldots, u) = \text{const}$. Można udowodnić, że funkcja x^y, a więc i funkcje n^x oraz y^m, gdzie m i n są stałymi, są obliczalne. Można również udowodnić, że funkcja

$$f(x, y, z, \ldots, u) = \begin{cases} 0, & \text{gdy } g(x, y, z, \ldots, u) = 0, \\ 1, & \text{gdy } g(x, y, z, \ldots, u) \neq 0, \end{cases}$$

gdzie $g(x, y, z, \ldots, u)$ jest jakąś funkcją obliczalną, jest obliczalna.

Funkcje obliczalne mają szerokie zastosowania (omówimy je w następnym paragrafie) i bogatą literaturę (patrz: Kleene [1952], Grzegorczyk [1961], Malcew, Davis, Péter). Nazwa funkcje obliczalne jest dla nich zupełnie naturalna. Wartości tych funkcji potrafimy naprawdę obliczyć.

Z rozważań podanych w zadaniu 3 tego paragrafu wynika, że dla każdej funkcji obliczalnej istnieje taka maszyna Turinga (zob. rozdział X par. 56), która, gdy wypiszemy argument na taśmie (np. w zapisie zero jedynkowym), wypisze po skończonej liczbie kroków wartość tej funkcji.

Streszczenie

Podaliśmy definicję funkcji obliczalnych (definicja 1). W następnym paragrafie powiemy o zastosowaniu funkcji obliczalnych.

Zadania

1. a) Udowodnić, że funkcje $f(x) = \text{const}$ jest obliczalna.

b) Udowodnić, że funkcja:

$$h(0) = 0,$$

$$h(x) = 1, \quad \text{gdy } x \neq 0,$$

jest obliczalna.

c) Wywnioskować z b), że funkcja:

$$f(x, y, z, \ldots, u) = \begin{cases} 0, & \text{gdy } g(x, y, z, u) = 0, \\ 1, & \text{gdy } g(x, y, z, u) \neq 0 \end{cases}$$

jest obliczalna.

d) Udowodnić, że funkcja

$$x \dot{-} y = \begin{cases} x-y, & \text{gdy } x > y, \\ 0, & \text{gdy } x \leqslant y \end{cases}$$

jest obliczalna.

e) Udowodnić, że funkcje $[x:y]$ oraz $[\sqrt{x}]$ są obliczalne.

Symbol $[x]$ (całość z x) czytamy jako: największa liczba całkowita $\leqslant x$.

f) Udowodnić, że funkcje $q(x, y)$ i $r(x, y)$ o wartościach całkowitych równe odpowiednio:

$$q(x, y) = \begin{cases} \text{iloraz dzielenia } x \text{ przez } y, \text{ dla } y \neq 0, \\ 0 \text{ dla } y = 0, \end{cases}$$

$$r(x, y) = \begin{cases} \text{reszta z dzielenia } x \text{ przez } y, \text{ dla } y \neq 0, \\ x \text{ dla } y = 0 \end{cases}$$

są obliczalne.

Zachodzi wtedy wzór

$$x = y \cdot q(x, y) + r(x, y)$$

i ponadto:

$$r(x, y) < y \text{ dla } y \neq 0.$$

2. Niech X będzie podzbiorem zbioru N, określmy funkcję:

$$f_X(x) = \begin{cases} 0, & \text{gdy } x \in X, \\ 1, & \text{gdy } x \notin X. \end{cases}$$

(Uwaga. Funkcja 1—$f_X(x)$ jest funkcją charakterystyczną zbioru X. Zob. R. VI, § 37).

a) Udowodnić, że gdy $X = \{x_1, \ldots, x_n\}$ jest skończonym zbiorem liczb naturalnych, wtedy funkcja $f_X(x)$ jest obliczalna.

b) Udowodnić, że gdy X jest zbiorem liczb naturalnych parzystych, wtedy funkcja $f_X(x)$ jest obliczalna.

c) Udowodnić, że gdy X jest zbiorem liczb nieparzystych, wtedy funkcja $f(x)$ jest obliczalna.

3. a) Mamy tabelki dwóch maszyn Turinga T_f i T_g służących do obliczania: wartości funkcji $f(x)$, dla zapisanego na taśmie argumentu x (maszyna T_f), oraz dla obliczania wartości funkcji $g(x, y)$ dla zapisanych na taśmie argumentów x i y (maszyna T_g). Podać tabelkę maszyny służącej do obliczania na taśmie wartości funkcji

$$h(x) = g(x, f(x))$$

dla zapisanego na taśmie argumentu x.

b) Podobnie mając tabelki maszyn T_f i T_g dla funkcji $f(x)$ oraz $g(x, y, z)$, podać tabelkę maszyny T_h, która dla wypisanej na taśmie pary argumentów x i y, oblicza wartości funkcji $h(x, y)$ otrzymanej z $f(x)$ i $g(x, y, z)$ przez schemat rekursji prostej podany w punkcie 2 tego paragrafu.

c) Mając tabelkę maszyny T_f służącej do obliczania wartości funkcji $f(x, y)$, podać tabelkę dla maszyny, do obliczania wartości funkcji

$$h(x) = \text{najmniejsze } y \text{ takie, że } f(x, y) = 0,$$

dającej po skończonej liczbie kroków wartość $h(x)$ (przy zapisanym na taśmie x), jeżeli tylko dla każdego x istnieje takie y, że $f(x, y) = 0$.

d) Posługując się wynikiem zadań a) b) c) i zadania 4 (5 lub 6) z § 63 tego rozdziału, uzasadnić, że dla każdej funkcji obliczalnej $f(x, y, z, \ldots, u)$ istnieje maszyna Turinga T_f obliczająca w skończonej liczbie kroków wartość funkcji $f(x, y, z, \ldots u)$, gdy mamy na taśmie wypisane wartości $x, y, z, \ldots, u$.

4. a) Udowodnić, że funkcji obliczalnych jest przeliczalnie wiele, tzn., że można wszystkie funkcje ustawić w ciąg $f_0, f_1, f_2, \ldots, f_n, \ldots$

Wskazówka. Udowodnić wpierw punkt *b)* tego zadania.

b) Udowodnić, że można wszystkie funkcje obliczalne jednej zmiennej ustawić w ciąg

$$h_0(x), h_1(x), h_2(x), \ldots$$

tak, że każda funkcja występuje w tym ciągu raz i tylko raz. Wywnioskować stąd, że istnieją funkcje, które nie są obliczalne.

c) Funkcja $h(x, n)$ taka, że dla każdego n $h(x, n) = h_n(x)$, gdzie $h_n(x)$ jest funkcją z ciągu podanego w zadaniu b) nazywa się *funkcją uniwersalną* dla funkcji obliczalnych jednej zmiennej. Uzasadnić, że funkcja $h(x, n)$ nie może być funkcją obliczalną.

Wskazówka. Funkcja $h(x, x)+1$ nie może być żadną z funkcji ciągu wypisanego pod b).

5. Klasa funkcji zawierająca funkcje $S(n)$, $0(x) = 0$, $I_1(x) = x$, $I_2(x, y) = y$ i zamknięta ze względu na: 1) składanie funkcji; 2) schemat rekursji prostej; 3) minimum, ale niekoniecznie efektywne, określone następująco:

$$h(x) = \begin{cases} \text{najmniejsze } y \text{ takie, że } f(x, y) = 0, \text{ gdy takie } y \text{ istnieje,} \\ \text{nieokreślona, gdy takiego } y \text{ nie ma;} \end{cases}$$

nazywa się *klasą funkcji częściowo obliczalnych.*

a) Uzasadnić, że klasa ta zawiera przeliczalnie wiele funkcji. Uzasadnić, że istnieją funkcje, które nie są częściowo obliczalne

b) Uzasadnić, że istnieje funkcja częściowo obliczalna, która nie może być uzupełniona w punktach nieokreśloności tak by otrzymać funkcję obliczalną.

6. a) Uzasadnić, że wartość funkcji częściowo obliczalnej może być wypisana w skończonej ilości kroków przez maszynę Turinga dla dowolnego argumentu dla którego taka wartość istnieje.

b) Uzasadnić hipotezę Turinga mówiącą, że dla każdej maszyny Turinga z symbolami | (pałeczka) pisanymi na taśmie, liczba pałeczek wypisanych przez maszynę po zatrzymaniu się, jako funkcja liczby pałeczek napisanych kolejno na taśmie przed uruchomieniem maszyny Turinga, jest funkcją częściowo obliczalną. (Zob. Kleene [1952] i inni cytowani w tekście).

§ 65. ZBIORY OBLICZALNE

W poprzednim paragrafie zdefiniowaliśmy funkcje obliczalne. Nazwa odzwierciedla fakt, że funkcje te są tak zdefiniowane, że potrafimy obliczyć ich wartości dla każdego argumentu. W tym paragrafie zajmiemy się zbiorami zer (tj. tych $x \in N$, dla których $f(x) = 0$) i zbiorami wartości funkcji obliczalnych.

Przyjmiemy następującą definicję:

DEFINICJA 1. *Zbiór* Z liczb naturalnych nazwiemy *obliczalnym*, jeżeli istnieje taka funkcja obliczalna $f(x)$, że

(1) $\qquad x \in Z$ wtedy i tylko wtedy, gdy $f(x) = 0$.

Definicja ta nazywa zbiory zer funkcji obliczalnych zbiorami obliczalnymi. Nazwa zbiory obliczalne wywołuje wiele skojarzeń, które wyjaśnimy w dalszej części tego paragrafu. Na razie jednak podamy przykłady i podstawowe własności zbiorów obliczalnych.

Zbiór N liczb całkowitych jest obliczalny, gdyż jest zbiorem zer funkcji $0(x) = 0$. Podobnie zbiór pusty Λ jest zbiorem obliczalnym, gdyż jest zbiorem zer funkcji obliczalnej $1(x) = 1$. Funkcja ta jest obliczalna gdyż $1(x) = S(0(x))$, funkcje zaś $S(x)$ — następnika oraz $0(x)$ — zerowa są obliczalne.

Zbiory skończone są zbiorami obliczalnymi. Udowodnimy to. Niech $Z_1 = \{x_1\}$ będzie zbiorem złożonym z jednego elementu. Funkcja

$$f(x) = x \dot{-} x_1 = \begin{cases} 0, & \text{gdy } x < x_1, \\ x - x_1, & \text{gdy } x \geqslant x_1 \end{cases}$$

jest obliczalna (zob. zad. 1 z poprzedniego paragrafu). Wynika stąd, że funkcja

$$g_1(x) = (x \dot{-} x_1) + (x_1 \dot{-} x)$$

jest obliczalna. Funkcja ta zeruje się w jednym jedynym punkcie $x = x_1$. A więc zbiór Z_1 jednoelementowy jest obliczalny. Jeżeli $Z = \{x_1, \ldots, x_n\}$ jest zbiorem n-elementowym, to jest zbiorem zer funkcji obliczalnej

$$g(x) = g_1(x) \cdot g_2(x) \cdot \ldots \cdot g_n(x),$$

gdzie

$$g_i(x) = (x \dot{-} x_i) + (x_i \dot{-} x) \quad \text{dla } i = 1, 2, \ldots, n.$$

Por. również zad. 2a) z poprzedniego paragrafu.

Zbiory liczb parzystych i nieparzystych są obliczalne. Funkcja obliczalna $r(x, z)$ (zob. zad. 1f z poprzedniego paragrafu) jest obliczalna; zeruje się tylko dla liczb parzystych. Funkcja $1 \dot{-} r(x, z)$ jest również obliczalna; zeruje się tylko dla liczb nieparzystych. Por. również zad. 2b) i c) z poprzedniego paragrafu. Wiele innych podzbiorów liczb naturalnych jest obliczalnych. Zachodzi następujące twierdzenie.

TWERDZENIE 1. *Zbiory obliczalne tworzą algebrę Boole'a względem operacji „∪" sumy, „∩" przekroju zbiorów i „′" uzupełnienia do zbioru N liczb naturalnych.*

Dowód. Jak już pokazaliśmy podzbiór N (zbiór pełny) oraz podzbiór pusty Λ są obliczalne. Stanowią one jedność oraz zero algebry. Niech Z_1 i Z_2 będą zbiorami obliczalnymi, $f_1(x)$ i $f_2(x)$ zaś funkcjami obliczalnymi, dla których zbiory te są zbiorami zer. Funkcje obliczalne $g(x) = f_1(x) \cdot f_2(x)$ oraz $f(x) = f_1(x) + f_2(x)$ będą się zerować:

funkcje $g(x)$ dla tych i tylko dla tych liczb naturalnych które należą do $Z_1 \cup Z_2$;

funkcja $f(x)$ dla tych i tylko tych liczb naturalnych, które należą do $Z_1 \cap Z_2$.

A więc zarówno $Z_1 \cup Z_2$ jak i $Z_1 \cap Z_2$ są zbiorami obliczalnymi. Jeżeli zbiór Z jest zbiorem obliczalnym, $f(x)$ zaś funkcją obliczalną, której zbiór zer jest zbiorem Z, to funkcja obliczalna $1 \dot{-} f(x)$ zeruje się dla tych i tylko tych liczb naturalnych dla których nie zeruje się funkcja $f(x)$. A więc zbiorem jej zer jest uzupełnienie Z' zbioru Z do zbioru N. Dowód twierdzenia został przeprowadzony w całości.

Twierdzenie 1 mówi, że zbiory obliczalne stanowią podalgebrę algebry Boole'a wszystkich podzbiorów zbioru N liczb naturalnych. Zajmiemy się teraz zagadnieniem, jak duża jest ta podalgebra — ile jest podzbiorów obliczalnych. Podamy mianowicie przykład zbioru, który nie jest obliczalny.

PRZYKŁAD. Funkcje obliczalne jednej zmiennej można ustawić w ciąg (por. zadanie 4a i 4b z poprzedniego paragrafu); niech będzie to ciąg

(2) $\qquad f_1(x), \quad f_2(x), \quad f_3(x), \quad \ldots$

Każda funkcja obliczalna występuje w tym ciągu jeden raz. Zbiór zer Z_i, (tzn. zbiór tych $x \in N$, dla których $f_i(x) = 0$) jest zbiorem obliczalnym. Otrzymujemy w ten sposób ciąg zbiorów obliczalnych

(3) $\qquad Z_0, Z_1, Z_2, Z_3, \ldots$

W ciągu tym stoją wszystkie zbiory obliczalne. Rzeczywiście, każdy zbiór obliczalny Z jest zbiorem zer jakiejś funkcji f, a funkcja ta występuje w ciągu (2). Odpowiadający jej zbiór obliczalny (zbiór Z) występuje w ciągu

(3). Zbiory z ciągu (3) mogą się powtarzać, gdyż różne funkcje obliczalne mogą mieć równe zbiory zer. Jeżeli z ciągu (3) usuniemy zbiory powtarzające się i będziemy numerować kolejno, to otrzymamy ciąg

$$U_0, U_1, U_2, U_3, \ldots, \tag{4}$$

w którym stoją wszystkie zbiory obliczalne i każdy tylko jeden raz.

Używając ciągu (4), określimy podzbiór U zbioru liczb całkowitych, który nie jest obliczalny. Określimy zbiór U następująco:

$$n \in U \text{ wtedy i tylko wtedy, gdy } n \notin U_n.$$

Udowodnimy, że zbiór U nie stoi w ciągu (4). Przypuśćmy, że $U = U_k$. Możliwe są dwa przypadki: albo $k \in U_k$, wtedy z definicji U jest $k \notin U = U_k$, albo $k \notin U_k$, wtedy z definicji U jest $k \in U$, stąd $U = U_k$. (Czytelnik zechce zastanowić się, gdzie korzystamy tu z założenia, że zbiory występujące w ciągu 4) są różne). Otrzymana sprzeczność pokazuje, że zbiór U nie może być równy żadnemu zbiorowi U_k dla $k = 0, 1, 2, \ldots$, a więc zbiór U nie jest obliczalny.

Dla czytelników znających elementy teorii mnogości podamy, że powyższe rozumowanie sprowadza się do faktu, że podzbiorów obliczalnych zbioru N jest przeliczalnie wiele, wszystkich zaś podzbiorów zbioru N jest nieprzeliczalnie wiele.

Zbiory obliczalne mają dzięki swoim własnościom wielkie znaczenie. Mianowicie dla każdego zbioru obliczalnego Z potrafimy efektywnie sprawdzić czy dowolna liczba naturalna należy do zbioru Z, czy też nie. Rzeczywiście ze zbiorem Z związana jest funkcja obliczalna $f(x)$, dla której zbiorem zer jest Z. Aby stwierdzić czy liczba naturalna n należy do Z czy nie, wystarczy obliczyć $f(n)$ (wartość funkcji $f(x)$ dla $x = n$). Jeżeli $f(n) = 0$, to $n \in Z$. Jeżeli $f(n) \neq 0$, to $n \notin Z$.

Zajmijmy się teraz zbiorami wartości funkcji obliczalnych.

DEFINICJA 2. Zbiory wartości funkcji obliczalnych nazywają się *zbiorami rekurencyjnie przeliczalnymi.*

Nazwa pochodzi od terminu „funkcje rekurencyjne" — używanego często na oznaczenie funkcji obliczalnych. Słowo „przeliczalne" występujące w tej nazwie oznacza, że elementy zbioru rekurencyjnie przeliczalnego można ponumerować, tzn. ustawić w ciąg, używając funkcji obliczalnej.

Jeżeli R jest zbiorem rekurencyjnie przeliczalnym, to znaczy to, że istnieje funkcja obliczalna $f(x)$ taka, że ciąg liczb naturalnych

$$f(0), \quad f(1), \quad f(2), \quad \ldots \tag{5}$$

(ciąg wartości funkcji $f(x)$) składa się ze wszystkich elementów zbioru R; oczywiście wartości w tym ciągu mogą się powtarzać. Funkcja $f(x)$ numeruje — czyli przelicza elementy zbioru R, pozwala ustawić je w ciąg. Zazwyczaj wygodnie jest uważać również zbiór pusty za zbiór rekurencyjnie przeliczalny.

Zbiór liczb naturalnych, zbiór liczb parzystych, zbiór liczb nieparzystych są rekurencyjnie przeliczalne. Są one zbiorami wartości funkcji obliczalnych odpowiednio $f_1(x) = x, f_2(x) = 2x, f_3(x) = 2x+1$. Zbiory skończone są również rekurencyjnie przeliczalne. Rzeczywiście, jeżeli $R = \{a_0, \ldots, a_n\}$ jest zbiorem skończonym, to łatwo zbudować funkcję obliczalną, której zbiorem wartości jest zbiór R. Na przykład będzie to funkcja $g(x)$ określona następująco:

$$\begin{aligned} g(0) &= a_0, \\ g(1) &= a_1, \\ &\ldots\ldots \\ g(n) &= a_n, \\ g(x) &= a_n, \quad x > n. \end{aligned}$$

Klasa zbiorów rekurencyjnie przeliczalnych jest bardzo obszerna. Zachodzi mianowicie

TWIERDZENIE 2. *Każdy zbiór obliczalny jest rekurencyjnie przeliczalny. Istnieją także zbiory rekurencyjnie przeliczalne, które nie są obliczalne.*

Przykładem takiego zbioru jest zbiór U określony w podanym poprzednio przykładzie.

Dowód tego twierdzenia wymaga znajomości większej ilości faktów dotyczących funkcji obliczalnych, których nie przedstawiono w tej książce. Dlatego odsyłamy czytelnika do literatury: Grzegorczyk [1] Malcew [6] Kleene [3] Davis [7]. Również bez dowodu podamy następne twierdzenie (dowód patrz np. Malcew [6] lub inne cytowane prace):

TWIERDZENIE 3. *Suma i przekrój skończonej liczby zbiorów rekurencyjnie przeliczanych są rekurencyjnie przeliczalne. Jeżeli zbiór R oraz jego uzupełnienie są rekurencyjnie przeliczalne, to R jest zbiorem obliczalnym.*

Z twierdzenia tego wynika w szczególności, że jeżeli zbiór rekurencyjnie przeliczalny R nie jest zbiorem obliczalnym, to jego uzupełnienie nie jest zbiorem rekurencyjnie przeliczalnym.

Z twierdzeń 2 i 3 możemy wywnioskować, że uzupełnienie U' zbioru U podanego w przykładzie nie jest zbiorem rekurencyjnie przeliczalnym.

Zbiorów rekurencyjnie przeliczalnych jest jak widzimy z twierdzenia 2 więcej niż zbiorów obliczalnych. Jednak nie wszystkie podzbiory zbioru N są rekurencyjnie przeliczalne. Zbiorów rekurencyjnie przeliczalnych jest przeliczalnie wiele, gdyż funkcji obliczalnych jest przeliczalnie wiele. Wszystkich podzbiorów zbioru N jest ilość nieprzeliczalna. Musi więc istnieć „bardzo dużo" podzbiorów zbioru N, które nie są zbiorami rekurencyjnie przeliczalnymi.

Streszczenie

Zbiór zer funkcji obliczalnej nazywa się zbiorem obliczalnym, zbiór wartości funkcji obliczalnej nazywa się zbiorem rekurencyjnie przeliczalnym. Zbiór pusty, zbiór N, zbiory skończone są obliczalne. Zbiory obliczalne tworzą algebrę Boole'a z operacjami sumy, przekroju i uzupełnienia zbioru do N. Dla zbioru obliczalnego Z, zagadnienie, czy liczba naturalna należy, czy też nie należy do Z, jest rozstrzygalne. Każdy zbiór obliczalny jest rekurencyjnie przeliczalny, ale istnieją zbiory rekurencyjnie przeliczalne, które nie są obliczalne. Nie każdy podzbiór zbioru N jest rekurencyjnie przeliczalny.

Zadania

1. Udowodnić, że jeżeli zbiór Z jest obliczalny, to funkcja

$$f_Z(x) = \begin{cases} 0, & \text{gdy } x \in Z, \\ 1, & \text{gdy } x \notin Z \end{cases}$$

jest obliczalna.

2. Podzbiór U produktu kartezjańskiego $N \times N$ (zbioru par liczb naturalnych) nazywa się obliczalny, jeżeli istnieje funkcja obliczalna $f(x, y)$ dwóch zmiennych taka, że para $(x, y) \in U$ wtedy i tylko wtedy, gdy $f(x, y) = 0$.

a) Udowodnić, że funkcja $z = C(x, y)$ obliczalna odwzorowująca wzajemnie jednoznacznie $N \times N$ na N (każda taka funkcja nazywa się *numeracją par*), odwzorowuje wzajemnie jednoznacznie zbiór podzbiorów obliczalnych zbioru $N \times N$ na zbiór podzbiorów obliczalnych N.

b) Udowodnić, że ustawienie par w ciąg (0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (0, 2), (3, 0), (2, 1) daje numerację $C(x, y)$, gdzie $C = C(x, y)$ oznacza numer miejsca pary (x, y) w ciągu. Pary (x, y) ustawiane są według wzrastającej sumy $x+y$, zaś dla równych sum — leksykograficznie.

c) *Numeracja* $C(x, y)$ z punktu b) nazywa się *kantorowską*. Udowodnić, że

$$C(xy) = \frac{(x+y+1)(x+y+2)}{2} + y + 1.$$

Podać przykłady innych numeracji.

d) Udowodnić, że zbiór podzbiorów obliczalnych zbioru $N \times N$ tworzy podalgebrę algebry Boole'a wszystkich podzbiorów zbioru $N \times N$.

3. Udowodnić, że jeżeli R jest zbiorem rekurencyjnie przeliczalnym, $f(x)$ zaś dowolną funkcją obliczalną, to zbiór S wartości funkcji $f(x)$ dla $x \in R$, jest rekurencyjnie przeliczalny.

4. Uzasadnić, że każdy zbiór nieskończony i rekurencyjnie przeliczalny jest zbiorem wartości funkcji obliczalnej przyjmującej dla różnych argumentów różne wartości (tzn. różnowartościowej).

5. Zbiór R jest rekurencyjnie przeliczalny wtedy i tylko wtedy, gdy istnieje taka funkcja obliczalna $f(x, y)$ dwóch zmiennych, że równanie $f(x, y) = 0$ ma rozwiązanie ze względu na y dla tych i tylko tych x, które należą do R.

6. a) Zbiór wartości funkcji częściowo obliczalnej (zob. zad. 5 z poprzedniego paragrafu) jest zbiorem rekurencyjnie przeliczalnym.

b) Dla danej funkcji częściowo obliczalnej zbiór tych x, dla których $f(x)$ nie jest określona, jest zbiorem rekurencyjnie przeliczalnym.

Wskazówka. Skorzystać z zadania 5.

§ 66. METODA ARYTMETYZACJI TEORII

W paragrafie tym zajmiemy się arytmetyzacją teorii, i zdefiniujemy pojęcie rozstrzygalności teorii.

Metoda arytmetyzacji polega na przypisaniu formułom teorii liczb naturalnych zwanych *numerami formuł*. Pozwala to mówić nie o własnoś-

ciach pewnych twierdzeń (czy zbiorów twierdzeń) teorii ale o własnościach ich numerów (czy też zbiorów numerów). Twierdzenia dotyczące teorii tzw. *meta-twierdzenia* można wtedy uważać za twierdzenia arytmetyki. Pozwala to wykorzystać aparat wprowadzony w poprzednich paragrafach do opisu i badania własności teorii.

Formuły teorii sformalizowanych są wyrażeniami zbudowanymi w pewien sposób z symboli występujących w alfabecie tej teorii. Na przykład dla rachunku zdań mamy symbole

(1) $$(,), \vee, \&, \sim, \Rightarrow, \equiv, p_1, p_2, p_3, \ldots$$

Dla elementarnej teorii półgrup mamy symbole

(2) $$(,), \vee, \&, \sim, \Rightarrow, \equiv, \mathrm{A}, \mathrm{E}, x_1, x_2, \ldots$$

elementarnego rachunku kwantyfikatorów i dwa symbole specjalne, „=" dla równości i „·" (kropka) dla znaku działania.

W obu wypadkach (rozpatrzonych tu nie dla jakiejś wyjątkowości, ale dla przykładu) mamy łatwą i prostą metodę efektywnego sprawdzenia czy jakieś wyrażenie jest formułą poprawnie zbudowaną, czy też nie. Daje to możliwość efektywnego ustawienia formuł języka teorii w ciąg

(3) $$\Phi_0, \quad \Phi_1, \quad \Phi_2, \quad \ldots,$$

tak by każda formuła występowała w ciągu raz i tylko raz.

Opiszemy tutaj jedną z takich metod zwaną *numeracją Gödla*, posługując się dla przykładu rachunkiem zdań. Symbolom (1) przyporządkujemy liczby będące kolejnymi liczbami pierwszymi, poczynając od trójki, a więc liczby 3, 5, 7, ... Pokazuje to tabelka:

symbol	(	)	$\vee$	&	$\sim$	$\Rightarrow$	$\equiv$	p_1	p_2	p_3	...
liczba	3	5	7	11	13	17	19	23	29	31	...

Dla każdego wyrażenia W określimy liczbę zwaną jego *numerem Gödlowskim*, oznaczaną przez Nr $[W]$. Jeżeli wyrażenie W stanowi ciąg symboli $s_i, \ldots, s_k$, tzn. $W = s_1 s_2 \ldots s_k$, gdzie symbolom $s_1, s_2, \ldots, s_k$ odpowiadają liczby $l_1, l_2, \ldots, l_k$ podane w tabelce, to numerem Gödlowskim

Nr $[W]$ wyrażenia będziemy nazywali liczbę

$$\mathrm{Nr}[W] = 2^{l_1} \cdot 3^{l_2} \cdot \ldots \cdot p_k^{l_k},$$

gdzie p_k jest k-tą liczbą pierwszą.

Tak więc numerem wyrażenia $p_1 \Rightarrow p_2$ będzie liczba Nr $[p_1 \Rightarrow p_2] = 2^{23} \cdot 3^{17} \cdot 5^{29}$, numerem wyrażenia p_3 będzie Nr $[p_3] = 2^{31}$ (a nie 31). Na odwrót, mając daną liczbę naturalną n możemy stwierdzić czy jest ona numerem Gödlowskim jakiegoś wyrażenia, czy też nie. Na przykład liczba 864 będzie numerem wyrażenia $2^5 \cdot 3^3$ oczywiście nie będącego formułą poprawnie zbudowaną, gdyż $864 = 2^5 \cdot 3^3 = \mathrm{Nr}\,)\,(\,)$. Podobnie liczba $2^{13} \cdot 3^3 \cdot 5^{23} \cdot 7^5 = \mathrm{Nr}\,[\sim(p)]$. Znajdowanie numerów Gödlowskich wyrażeń i badanie czy liczba jest numerem Godlowskim jakiegoś wyrażenia, jest procesem efektywnie wykonalnym, składającym się z pewnej ilości nieskomplikowanych operacji, aczkolwiek absurdalnie pracochłonnym, gdyż numery Gödlowskie nawet prostych formuł, są olbrzymimi liczbami. Metoda numeracji Gödla i jej znaczenie opisane jest prosto i przejrzyście w książce Nagela i Neumana.

Metoda numeracji Gödlowskiej umożliwia ustawienie wszystkich formuł teorii w ciąg (3) tak, że każda formuła występuje w ciągu raz i tylko raz. Dla konstrukcji tego ciągu wystarcza wypisywać kolejno wyrażenia, według wzrastających numerów Gödlowskich, i wykreślać te, które nie są formułami poprawnie zbudowanymi.

Ciąg (3) pozwala określić nową numerację pod pewnymi względami stosowniejszą od Gödlowskiej. Mianowicie formule Φ przyporządkujemy numer miejsca na którym stoi ona w ciągu (3). Numer ten oznaczymy symbolem $C[\Phi]$ i nazywać będziemy w dalszym ciągu *numerem formuły* Φ. A więc

$$C[\Phi_n] = n.$$

Numeracja ta jest o tyle korzystniejsza od Gödlowskiej, że daje odwzorowanie zbioru formuł na zbiór liczb naturalnych; każda liczba naturalna jest numerem $C[\Phi]$ jednoznacznie wyznaczonej formuły Φ i każda formuła ma jednoznacznie wyznaczony numer $C[\Phi]$. Co więcej numer każdej formuły może być efektywnie wyznaczony. Aby znaleźć numer $C[\Phi]$ formuły Φ należy znajdować kolejne formuły w ciągu (3) tak długo, aż nie

natrafimy na formułę Φ, ilość formuł ją poprzedzających (tzn. numer miejsca formuły Φ) będzie jej numerem. Podobnie dla każdej liczby naturalnej można efektywnie wyznaczyć formułę Φ o tym numerze tzn. formułę Φ_n. Wystarczy znaleźć $n+1$ początkowych wyrazów ciągu (3). Ostatni będzie formułą o numerze n.

Zreasumujemy otrzymane wyniki. Przypisaliśmy każdej formule Φ teorii jednoznacznie wyznaczoną liczbę naturalną $C[\Phi]$ zwaną numerem tej formuły i każdej liczbie naturalnej n — formułę Φ, której numer $C[\Phi] = n$, tak że przejście od formuły do numeru i od numeru do formuły są efektywne. Pozwala to zamiast o formułach mówić o liczbach naturalnych, zamiast o zbiorach formuł mówić o zbiorach liczb naturalnych, zamiast o operacjach na formułach mówić o funkcjach określonych dla liczb naturalnych o wartościach będących liczbami naturalnymi. Postępowanie takie nazywa się *arytmetyzacją teorii.*

Na przykład zamiast mówić o regule odrywania prowadzącej od formuły Φ i formuły χ postaci $\Phi \Rightarrow \Psi$, do formuły Ψ, możemy mówić o funkcji $f(x, y) = z$, określonej na liczbach naturalnych o wartościach naturalnych, takiej że: jeżeli $x = C[\Phi]$, $y = C[\Theta]$, formuła zaś Θ jest postaci $\Phi \Rightarrow \Psi$ to $z = f(x, y) = C[\Psi]$. Funkcja ta nie jest wszędzie określona np. jeśli formuła Θ nie jest postaci $\Phi \Rightarrow \Psi$ dla jakiegoś Ψ, to aby określić $f(x, y)$ można położyć np. $f(x, y) = 0$. Można sprawdzić, że tak określona funkcja $f(x, y)$ jest funkcją obliczalną. Podobnie dla reguły podstawiania prowadzącej od formuły $\Phi(p_1, \ldots, p_n)$ do formuły $\Psi = \Phi(\Theta, p_2, \ldots, p_n)$ funkcja $z = g(x, y)$ określona dla liczb naturalnych, taka że dla $x = C[\Phi]$ oraz $y = C[\Theta]$ wartość $z = g(x, y) = C[\Psi]$, jest obliczalna. Inne reguły wnioskowania używane w rachunku kwantyfikatorów, czy też w teoriach elementarnych dają w wyniku takiej arytmetyzacji funkcje obliczalne(1).

Aksjomatów specyficznych teorii jest na ogół liczba skończona. Zbiór ich numerów jako zbiór skończony jest obliczalny. Zachodzi następujące twierdzenie:

TWIERDZENIE 1. *Jeżeli teoria ma skończoną liczbę aksjomatów specyficznych i reguł wnioskowania i jeżeli reguły wnioskowania są efektywne,*

(1) Jeżeli tylko liczba symboli (stałych teorii i zmiennych) jest przeliczalna, tzn. symbole dadzą się ustawić w ciąg, gdyż wtedy można określić numerację Gödlowską Nr $[\Phi]$ i numerację $C[\Phi]$ dla formuł teorii.

tzn. odpowiednie funkcje określone na numerach formuł są obliczalne, to zbiór numerów twierdzeń teorii jest rekurencyjnie przeliczalny(1).

Precyzyjny dowód tego twierdzenia jest oczywiście bardzo skomplikowany. Sens jego jest jednak bardzo prosty. Idzie o to, że mamy efektywną metodę wypisywania kolejno twierdzeń teorii, tak by każde twierdzenie było kiedyś wypisane. Po prostu wypiszemy najpierw aksjomaty, potem do każdego aksjomatu stosujemy kolejno reguły wnioskowania i wypisujemy kolejne wyniki, następny krok to stosowanie znowu wszystkich reguł wnioskowania do już otrzymanych twierdzeń, itd. Oczywiście po n-krotnym wykonaniu takiego postępowania wypiszemy wszystkie twierdzenia, których dowód sformalizowany ma długość n.

Jeżeli zbiór numerów twierdzeń teorii jest zbiorem obliczalnym to mówimy, że teoria jest rozstrzygalna. Dla każdej formuły istnieje wtedy efektywna metoda rozstrzygnięcia w skończonej liczbie kroków czy formuła jest twierdzeniem teorii, czy też nie. Dla formuły Φ musimy obliczyć jej numer $C[\Phi]$, co jest efektywnie wykonalne, oraz stwierdzić czy $C[\Phi]$ należy do zbioru numerów twierdzeń teorii, czy też nie. A jak wynika z rozważań poprzedniego paragrafu, w przypadku zbiorów obliczalnych potrafimy to efektywnie stwierdzić.

Naszkicowaliśmy w tym paragrafie elementy metody arytmetyzacji teorii. Polega ona na efektywnym przyporządkowaniu numerów formułom poprawnie zbudowanych teorii. Wtedy różne twierdzenia dotyczące teorii przechodzą na pewne twierdzenia arytmetyki. Większość wyników dotyczących rozstrzygalności teorii została uzyskana dzięki metodzie arytmetyzacji.

Streszczenie

Opisaliśmy efektywną metodę przypisywania formułom numerów. Stwierdziliśmy, że opisane przyporządkowanie jest efektywne. W związku z efektywną możliwością dowodzenia twierdzeń teorii zbiór numerów

(1) Żądamy oczywiście by liczba stałych teorii i liczba zmiennych teorii były przeliczalne, tak by można było wprowadzić numerację. Założenia o skończoności zbioru aksjomatów i reguł wnioskowania można osłabić; kwestie te omówione są w książce Tarskiego, Robinsona i Mostowskiego oraz w cytowanych już podręcznikach.

twierdzeń teorii jest zbiorem rekurencyjnie przeliczalnym. W przypadku gdy zbiór ten jest zbiorem obliczalnym mówimy, że teoria jest rozstrzygalna.

Zadania

1. Uzasadnić, że funkcja $f(n) = p_n$ (p_n jest n-tą z kolei liczbą pierwszą) jest funkcją obliczalną.

2. Rozpatrujemy rachunek zdań i numerację opisaną w tekście

a) Niech $\Phi = (p_1) \,\&\, (p_2 \Rightarrow p_3)$.
Obliczyć $\mathrm{Nr}[\Phi]$.

b) Niech wyrażenie Φ będzie zdaniem $(p_1 \equiv p_2) \vee p_1$.
Obliczyć $\mathrm{Nr}\,[\Phi]$.

c) Znaleźć wyrażenie Φ, jeżeli wiadomo, że

$$\mathrm{Nr}[\Phi] = 2^3 \cdot 3^{23} \cdot 5^5 \cdot 7^7 \cdot 11^3 \cdot 13^{11} \cdot 17^3 \cdot 19^{11} \cdot 23^{23} \cdot 29^5 \cdot 31^5.$$

d) Czy liczba $2^4 \cdot 3^7 \cdot 5^5 \cdot 7^{11}$ jest numerem jakiejś formuły?

e) Czy wyrażenie Φ, którego numer $\mathrm{Nr}[\Phi] = 2^3 \cdot 3^{29} \cdot 5^{23} \cdot 7^{17} \cdot 11^3 \cdot 13^5$ jest formułą poprawną teorii?

3. Rozpatrzymy rachunek zdań i numerację opisaną poprzednio

a) Uzasadnić, że funkcja

$$h(n) = \begin{cases} 0, & \text{gdy } n \text{ jest numerem Gödlowskim formuły poprawnej} \\ 1, & \text{w przeciwnym przypadku} \end{cases}$$

jest obliczalna.

b) Wypisać aksjomaty S_1-S_3 rachunku zdań z paragrafu 14 rozdział II, str. 51 przy użyciu wszystkich nawiasów i obliczyć ich numery Gödlowskie.

c) Uzasadnić, że funkcja $f(n, m, k)$ równa 0, gdy istnieją formuły poprawne Φ i Ψ takie, że $\mathrm{Nr}[\Phi] = n$, $\mathrm{Nr}[\Psi] = k$, $\mathrm{Nr}[(\Phi) \Rightarrow (\Psi)] = m$, równa zaś 1 w przypadku przeciwnym, jest obliczalna.

d) Uzasadnić, że funkcja $g_i(n, m) = k$, gdzie i jest numerem zmiennej (np. p_i) równa 0, gdy istnieje takie wyrażenie poprawne Φ i χ, że $\mathrm{Nr}[\Phi] = n$, $\mathrm{Nr}[\chi] = m$, k zaś jest numerem wyrażenia Ψ powstałego z Φ przez podstawienie za zmienną p_i wyrażenia χ, równa zaś 1 w przypadku przeciwnym, jest obliczalna.

(Uwaga. Gdy Φ nie zawiera zmiennej p_i, to po podstawieniu Φ nie ulegnie zmianie).

4. Rozpatrujemy rachunek zdań i numerację opisaną poprzednio.

a) Uzasadnić, że dla dwóch formuł Φ, Ψ zachodzi równoważność

$$\mathrm{Nr}[\Phi] < \mathrm{Nr}[\Psi] \quad \text{wtedy i tylko wtedy, gdy } C[\Phi] < C[\Psi].$$

b) Formuła Φ_0 z ciągu (3) jest formułą o najmniejszym numerze Gödlowskim. Znaleźć tę formułę (zdaniem autorów $\Phi_0 = p_1$, $\Phi_2 = p_2$, $\Phi_3 = p_3$ dalej nie prowadziliśmy konstrukcji ciągu (3)).

c) Udowodnić, że jeżeli część Ψ formuły Φ sama jest formułą, to $C[\Psi] < C[\Phi]$.

5. Rozpatrujemy rachunek zdań i numerację opisaną w tekście.

a) Udowodnić, że funkcja $d(n)$ równa liczbie symboli w wyrażeniu, gdy n jest numerem wyrażenia poprawnego, lub 0, gdy n nie jest numerem wyrażenia poprawnego, jest obliczalna.

b) Uzasadnić, że funkcja $z(n)$ równa liczbie zmiennych wolnych w wyrażeniu Φ, gdy n jest numerem Nr $[\Phi]$ wyrażenia Φ będącego formułą poprawnie zbudowaną, równa zaś 0 w przypadku przeciwnym, jest obliczalna.

6. a) Udowodnić, że zbiór numerów $C[\Phi]$ tautologii rachunku zdań jest obliczalny.

b) Udowodnić, że funkcja $h(n)$ równa 0, gdy n jest numerem tautologii rachunku zdań, równa zaś 1 w przypadku przeciwnym, jest obliczalna.

7. a) Podać efektywną metodę numeracji wyrażeń poprawnych rachunku kwantyfikatorów, zawierających jeden predykat $\mathrm{P}(x_1, \ldots, x_n)$.

8. Opierając się na twierdzeniu 1 udowodnić, że jeżeli teoria jest zupełna, tzn. dla każdego zdania Φ, albo Φ albo $\sim \Phi$ jest twierdzeniem teorii, to teoria jest rozstrzygalna. Zauważyć, że funkcja $y = h(x)$ prowadząca od $x = C[\Phi]$ do $y = h(x) = C[\sim \Phi]$ jest obliczalna. Jeżeli T jest zbiorem numerów twierdzeń teorii, to $h(T)$ będzie zbiorem rekurencyjnie przeliczalnym (por. zad. 3 z poprzedniego paragrafu). Wobec tego, że $h(T)$ jest uzupełnieniem zbioru T do zbioru N liczb naturalnych, wystarczy skorzystać z twierdzenia 3 z poprzedniego paragrafu.

9. Liczby przyporządkowane symbolom elementarnej teorii półgrup podane są w tabelce

symbol	·	=	(	)	∨	&	∼	⇒	≡	A	E	x_1	x_2	x_3
liczba	3	5	7	11	13	17	19	23	29	31	37	41	43	47

a) Znaleźć numer $\mathrm{Nr}[\Phi]$ formuły Φ równej $\mathrm{A}x_1(x_1 = x_2)$.

b) Czy wyrażenie $2^{41} \cdot 3^5 \cdot 5^{41}$ jest formułą?

c) Znaleźć formułę Φ taką, że $C[\Phi] = 0$.

d) Uzasadnić, że funkcja $f(n) = m$ określona następująco:

jeżeli $n = \mathrm{Nr}[W]$, gdzie $W = x_i$ dla $i = 1, 2, \ldots$, to $m = f(n) = \mathrm{Nr}[V]$, gdzie V jest wyrażeniem $W \cdot x_j$, zaś $f(n) = 0$ w przypadku przeciwnym

jest obliczalna.

e) Udowodnić, że podstawienie za zmienną x_i w formule termu $\tau(x_1, \ldots, x_n)$ zbudowanego ze znaku działania „·" prowadzi do funkcji obliczalnej określonej na numerach wyrażeń.

10. Niech alfabet składa się z liter $X_1, \ldots, X_s$. Za wyrażenia będziemy uważać skończone ciągi symboli $W = X_{a_k} X_{a_{k-1}} \ldots X_{a_1} X_{a_0}$ oraz symbol Λ, czyli elementy półgrupy wolnej (zob. rozdz. VIII, § 48) generowanej przez $X_1, \ldots, X_s$. Symbolom

$X_1, X_2, \ldots, X_s$ przyporządkujemy liczby $1, 2, \ldots, s$ i określimy numery wyrażeń następująco:

$$C[\Lambda] = 0,$$

$$C[X_{a_k} X_{a_{k-1}} \ldots X_{a_1} X_{a_0}] = a_0 + a_1 s + \ldots + a_{k-1} s^{k-1} + a_k s^k.$$

a) Udowodnić, że każda liczba naturalna jest numerem jednego i tylko jednego wyrażenia.

b) Udowodnić, że funkcja $Z(k, l) = m$ określona następująco:

jeżeli k jest numerem wyrażenia W, l zaś numerem wyrażenia V, to $m = Z(k, l)$ jest numerem wyrażenia $W * V$ (konkatenacji W i V), jest obliczalna.

Wskazówka. Udowodnić, że $C[W * X_i] = fC[W] + i$.

§ 67. TEZA CHURCHA

Rozważania tego rozdziału nie będą kompletne, gdy nie wyjaśnimy jaki jest związek pojęcia obliczalności wprowadzanego w tym rozdziale, z pojęciami efektywności wprowadzonymi w rozdziałach IX-XII. W rozdziałach tych określiliśmy różnego rodzaju pojęcia efektywnej metody przeprowadzania różnych operacji. Zawsze, zarówno w przypadku maszyn Turinga, algorytmów Markowa, czy innych pokrewnych pojęć, określony był jasno i wyraźnie przepis postępowania pozwalający z danych wyjściowych otrzymywać wynik. Postępowanie opisane np. przez maszynę Turinga, czy też np. algorytm Markowa dawało się zawsze wykonać niezależnie od umiejętności i inwencji wykonującego operacje. Podobnie jest w przypadku funkcji obliczalnych. Dla funkcji obliczalnej jest jasno określony przepis pozwalający z danych liczb naturalnych (danych wyjściowych) otrzymywać wartość funkcji (wynik).

Postępowanie to wydaje się na pierwszy rzut oka mało ogólne. Nie dość, że operujemy na liczbach naturalnych, to jeszcze stosujemy do nich bardzo proste operacje: następnika, składanie funkcji, operację rekursji oraz minimum efektywnego.

Dzięki metodzie efektywnej numeracji wyrażeń stwierdzamy, że operowanie liczbami naturalnymi zamiast wyrażeniami nie zmniejsza ogólności rozważań. Istnieją zresztą prace definiujące pojęcia obliczalności wprost dla napisów utworzonych z dowolnych symboli bez mówienia o liczbach naturalnych (zob. Malcew).

Zajmiemy się więc kwestią ogólności definicji pojęcia obliczalności. Pojęcie to pochodzi w zasadzie od Gödla jest jednak wynikiem bogatego, aczkolwiek krótkiego rozwoju historycznego. O kolejnych etapach rozwoju tego pojęcia można się zorientować z literatury (Grzegorczyk [1961], Malcew). Od najwęższych klas funkcji, dopuszczających poza operacją następnika operację składania i rekursji prostej, tzw. funkcji pierwotnie rekurencyjnych — poprzez klasy pośrednie, wykształciło się pojęcie funkcji obliczalnych, przez dopuszczenie jeszcze operacji minimum efektywnego, takie jakie przyjmujemy obecnie.

Każde postępowanie opisane za pomocą funkcji obliczalnej jest efektywne. Postępowania dające się opisać za pomocą maszyn np. Turinga czy innych, czy też za pomocą algorytmów np. Markowa, Posta czy innych, są również efektywne.

Już w roku 1936, A. Church wysunął tezę, że każde postępowanie „efektywne" daje się opisać za pomocą funkcji obliczalnych.

Hipoteza ta nosi nazwę *tezy Churcha*. Udowodnić jej oczywiście nie można bo pojęcie efektywności nie jest sprecyzowane. Można by ją jednak obalić gdybyśmy znaleźli jakieś proste postępowanie, które bylibyśmy skłonni uznać za efektywne, nie dające się opisać za pomocą funkcji obliczalnych.

Teza Churcha nie została jednak dotąd obalona. Wszystkie stworzone dotąd definicje postępowań efektywnych, czy to używające pojęcia maszyny Turinga (względnie innych maszyn), czy to używające algorytmów Markowa, systemów Tuego, Posta czy innych dają się opisać przez funkcje obliczalne. Dla obliczalnych zbiorów danych zbiór wyników jest zawsze zbiorem rekurencyjnie przeliczalnych. Konkretne wyniki znajdzie czytelnik w książkach Kleene'a [1952], Malcewa, Davisa.

Świadczy to o tym, że pojęcie obliczalności daje jak dotąd wystarczająco dobrą definicję pojęcia efektywnej metody rozwiązywania jakichś problemów. Jak dotąd o efektywności postępowania mówiło się w terminach funkcji obliczalnych. Nowsze prace operują pojęciem obliczalności opartym o maszyny Turinga (zob. Kleene'a [1952], Davisa, Malcewa), jest to w praktyce częstokroć wygodniejsze — dowody twierdzeń są znacznie krótsze i mniej skomplikowane. Bogato rozwinięta jest teoria efektywności oparta o algorytmy Markowa (zob. Markowa i Malcewa).

Literatura cytowana

Arbib, M., *Brains, Machines and Mathematics*, New York 1964.

Bar-Hillel, Y., Perles, M., Shamir, E., *On formal properties of Simple phrase structural grammars*. Applied Logic Branch, The Hebrew University of Jerusalem, Technical Report nr 4, 1964.

Bar-Hillel, Y., Gaifman, C., Shamir, E., *On categorial and phrase structural grammars*, The Bulletin of the Research Council of Israel, 9F, nr 1, June 1960.

Bernays, P., Hilbert, D., *Grundlagen der Mathematik*, Berlin 1934-39.

Birkhoff, G., *Lattice Theory*, New York 1948.

Birkhoff, G., Mac Lane, S., *Przegląd algebry współczesnej*, Warszawa 1963.

Borkowski, L., Suszko, R., *Wstęp do teorii mnogości i logiki matematycznej*, Warszawa 1966.

Cejtin, G. S. (Цейтин, Г. С.), *Ассоциативное иссчисление с неразрешимой проблемой эквивалентности*, Д. А. Н СССР, t. 107 (1956), str. 370-371.

Chomsky, N., Miller, G. A., *Introduction to the formal analysis of natural languages*, Handbook of mathematical psychology, New York — London 1963, str. 269—322.

Chomsky, N., Miller, G. A., *Formal properties of grammar*, Handbook of mathematical psychology, New York—London 1963, str. 323—418.

Chomsky, N., Miller, G. A., *Finitary models of language users*, Handbook of mathematical psychology, New York—London 1963, str. 419-491.

Cluskey, E. J., *A survey of switching circuits theory*, New York 1962.

Davis, M., *Computability and Unselvability*, New York 1958.

Elgot, C. C., Robinson, A., *Random — Access Stored Program Machines*, Journal of ACM, vol. 11, No 4 (1964).

Fraenkel, A., *Abstract Set Theory*, Amsterdam 1953.

Głuszkow, W. M., *Wstęp do cybernetyki*, Warszawa 1967.

Ginsburg, S., *Mathematical Theory of Context free-languages*, New York 1966.

Goodstein, R. L., *Boolean algebra*, New York 1963.

Grzegorczyk, A., *Logika popularna*, Warszawa 1955.

Grzegorczyk, A., *Zarys logiki matematycznej*, Warszawa 1961.

Hall, M., *The theory of groups*, New York 1959.

Hao Wang, *A Survey of Mathematical Logic*, Amsterdam 1963.

Hermes, H., *Einführung in die Verbandstheorie*, Berlin 1955.

Hermes, H., *Aufzählbarkeit, Entscheidbarkeit, Berechnerbarkeit*, Berlin 1967.

Hohn, F. E., *Applied Boolean algebra*, New York 1960,

Hilbert, D., Ackerman, N., *Grundzüge der theoretischen Logik*, wyd. 2, New York 1946.

Jacobson, N., *Lectures in abstract algebra*, t. I, Amsterdam 1953.

Kalmár, L., *Colloquium on the Foundations of mathematics, mathematical machines and their applications*, Budapest 1962.

Kleene, S. C., *Introduction to metamathematics*, Amsterdam 1952.

Kleene, S. C., *Representations of eventes in nerve nets and finite automata*, Automata Studies, 1956.

Kuratowski, K., *Wstęp do teorii mnogości i topologii*, Warszawa 1955.

Kuratowski, K., Mostowski, A., *Teoria mnogości*, Warszawa 1952.

Kobrynskij, N. E., Trachtenbrot, B. A. (Кобрынский, Н. Е., Трахтенброт, Б. А.), *Введение в теорию конечных автоматов*, Москва 1962.

Kurosz, A. G., (Курош, А. Г.), *Теория групп*, изд. 3, Москва 1963.

Kurosz, A. G., *Algebra ogólna*, Warszawa 1965.

Lapin, P. C., (Лапин, П. Ц.), *Полугруппы*, Москва 1960.

Lyndon, R. C., *O logice matematycznej*, Warszawa 1968.

Malcew, A. I. (Мальцев, А. И.), *Алгорифмы и рекурсивные функции*, Москва 1965.

Markow, A. A. (Марков, А. А.), *Теория алгорифмов*, Труди Стеклова 42 (1954).

Mendelsohn, *Introduction to mathematical logic*, New York 1964.

Miller, R. E., *Switching Theory*, New York 1962.

Mostowski, A., *Logika matematyczna*, Warszawa 1948.

Mostowski, A., Stark, M., *Algebra wyższa*, t. III, Warszawa 1964.

Mostowski, A. W., *Algebry Boole'a i ich zastosowania*, Warszawa 1964.

Moszner, Z., *O teorii relacji*, Warszawa 1967.

Nagel, E., Newman, J. R., *Twierdzenie Gödla*, Warszawa 1966.

Nowikow, P. S. (Новиков, П. С.), *Об алгорифмической неразрешимости проблемы тождества слов в теории групп*, Труди Математического Института Стеклова, 44 (1955).

Nowikow, P. S. (Новиков, П. С.), *Элементы математической логики*, Москва 1959.

Pawlak, Z., *Apllication of Two- valued Lattices to the Realization of Many-valued Truth-tables*, Biuletyn PAN, t. 9, nr 11, seria nauk technicznych, 1961.

Péter, R., *Recursive Funktionen*, Budapest 1951.

Pogorzelski, W., Słupecki, J., *O dowodzie matematycznym*, Warszawa 1962.

Post, E. L., *Finite Combinatory Process Formulation* 1, The Jurnal of Symbolic Logic 1 (1936), str. 103—105.

Post, E. L., *Formal Reductions of the General Combinatorial Decision Problem*, American Journal of Mathematics 65, No 2 (1943), str. 197—215.

Post, E. L., *A variant of a Recursively unsolvable Problem*, Bulletin of American Mathematical Society 52, No 4 (1946), str. 264—268.

Post, E. L., *Recursive Unsolvability of a Problem of Thue*, The Journal of Symbolic Logic 12 (1947), str. 1—11.

Rabin, O., Scott, D., *Finite Automata and their decision problems*, IBM Journal of research and Development, 3 (1959), str. 115—125.
Rasiowa, H., *Wstęp do logiki matematycznej i teorii mnogości*, Wrocław 1966.
Rasiowa, H., *Wstęp do matematyki współczesnej*, Warszawa 1969.
Rasiowa, H., Sikorski, R., *Mathematics of Metamathematics*, Warszawa 1963.
Roginskij, W. N. (Рогинский, В. Н.), *Построение релейных схем управления*, Москва 1964.
Słupecki, J., Borkowski, L., *Elementy logiki matematycznej i teorii mnogości*, Warszawa 1963.
Sierpiński, W., *Wstęp do teorii mnogości i topologii*, Warszawa 1965.
Sikorski, R., *Boolean algebras*, Berlin 1960.
Tarski, A., *Introduction to Logic and to the Methodology of Deductive Sciences*, New York 1965.
Thiele, H., *Wissenschaftstheoristische untersachungen in algorytmische Sprachen*, Berlin 1966.
Thue, A., *Probleme über Veränderungen von Zeichenreihen nach gegenübens Regeln*, Skr. Vidensk. Selsk. Kristiana I, 10 (1914).
Turing, A. M., *On Computable Numbers, with an application to the Entscheidungs problem*, Proceedings of the London Mathematical Society, Ser. 2. 42 (1936), str. 230—265.
Whitehead, A. N., Russell, B., *Principia Mathematicae*, Cambridge 1910, przedruk 1957.
Whitesitt, J., Eldon, *Boolean algebra and its applications*, London 1961.

Wykaz oznaczeń

p, q, r	zmienne zdaniowe	22
$p \vee q$	alternatywa (suma logiczna) zdań p i q	22
$p \& q$	koniunkcja (iloczyn logiczny) zdań p i q	22
$p \equiv q$	równoważność zdań p i q	22
$p \Rightarrow q$	implikacja zdań p i q	22
$\sim p$	negacja (zaprzeczenie) zdania p	22
0	fałsz	27
1	prawda	27
$\vdash$	symbol oznaczający tautologię	32
$p \not\equiv q$	różnica symetryczna zdań p i q	38
$p \mid q$	jednoczesne zaprzeczenie zdań p i q	38
$p \setminus q$	dyzjunkcja Sheffera	40
P, Q, R	zmienne predykatywne	59
$P(x), Q(x, y), R(x, y, z)$	predykaty	59
Ax	kwantyfikator ogólny	65
Ex	kwantyfikator szczegółowy	65
$\{a_1, \ldots, a_n\}$	zbiór złożony z elementów $a_1, \ldots, a_n$	65
$x \mid y$	x dzieli y	67
$a \in A$	element a należy do zbioru A	106
$a \notin A$	element a nie należy do zbioru A	108
1	zbiór pełny	107
0	podzbiór pusty	109
A, B, C	zbiory	109
$A \cup B$	suma zbiorów A i B	110
$A \cap B$	przekrój zbiorów A i B	110
A'	dopełnienie zbioru A	111
$A \subset B$	stosunek (relacja) inkluzji między zbiorami A i B	111
(x, y)	para uporządkowana	122

$X \times Y$	iloczyn (produkt) kartezjański zbiorów X i Y	122
$x \mathrm{R} y$	przedmiot x jest w relacji R z przedmiotem y	124
$\mathrm{R} \cup \mathrm{S}$	suma relacji R i S	126
$\mathrm{R} \cap \mathrm{S}$	iloczyn relacji R i S	126
R'	negacja relacji R	126
$\mathrm{S} \cdot \mathrm{R}$	złożenie (iloczyn względny) relacji R i S	126
$=$	relacja równości	127
rwl	relacja równoliczności	129
$\equiv$	relacja równoważności	130
$[x]$	klasa abstrakcji elementu x	130
$x \underset{k}{\equiv} y$	x i y należą do tej samej klasy abstrakcji	131
$x \mid y$	relacja podzielności (x dzieli y)	134
$\prec$	uporządkowanie ~~klasy~~leksykograficzne	136
R^*	relacja odwrotna do R	138
$a \cup b$	kres górny pary (a, b)	140
$a \cap b$	kres dolny pary (a, b)	140
$f(x)$	wartość funkcji dla argumentu x	143
f^{-1}	funkcja odwrotna ~~rzeczywista~~	144
$f(X)$	obraz zbioru X przez funkcję f	145
$f_A(x)$	obcięcie funkcji f do zbioru A	145
$f^*_X(x)$	przedłużenie funkcji f na zbiór X	145
fg	złożenie funkcji f i g	147
$T(B)$	zbiór wszystkich atomów elementu B	168
M_n	zbiór macierzy kwadratowych ustalonego stopnia	175
$\mathbf{A}, \mathbf{B}, \mathbf{C}$	macierze kwadratowe	175
$\mathbf{A} \cdot \mathbf{B}$	iloczyn (złożenie) macierzy	175
$\mathbf{E}$	macierz jednostkowa	176
$\mathbf{O}$	macierz zerowa	176
$P/\sim$	półgrupa ilorazowa półgrupy P względem relacji kongurencji „$\sim$"	191
$\Pi(X)$	zbiór wszystkich ciągów dowolnej skończonej długości	194
Λ	ciąg pusty o długości 0	197
$A = \{a_1, \ldots, a_n\}$	alfabet	206
$a_1, \ldots, a_n$	litery alfabetu	206
P, Q, R	słowa alfabetu	206
$\varnothing$	słowo puste	206

A^*	zbiór wszystkich słów w alfabecie A	206
$d(P)$	długość słowa P	206
$P \subset Q$	słowo P zawiera się w słowie Q	206
$P = Q$	słowa sąsiednie	208
$T \sim S$	słowa równoważne	208
K_s	zbiór słów końcowych słownika	224
$Q \underset{G}{\Rightarrow} P$ $Q \to P$	zdanie P wynika bezpośrednio ze zdania Q w gramatyce G	224
T	taśma maszyny Turinga	230
S	urządzenie sterujące maszyny Turinga	230
s_i	aktualny stan czynny maszyny Turinga	231
$\bar{s}_i$	stan bierny maszyny Turinga	231
P	przejście do obserwowania prawej klatki	232
L	przejście do obserwowania lewej klatki	232
N	niezmienienie obserwowanej klatki	232
$\to a_n$	wpisanie do obserwowanej kratki symbolu a_n	237
$(G_0/i_0, \ldots, a_n/i_n)$	przejście warunkowe	237
Stop	przerwanie działania maszyny	238
$M_i(P)$	słowo końcowe wyprodukowane przez maszynę M_i ze słowa początkowego P	253
P	taśma programowa maszyny uniwersalnej Posta	254
R	taśma robocza maszyny uniwersalnej Posta	254
$p_0, p_1, \ldots, p_n$	stany związane z taśmą P	255
$r_0, r_1, \ldots, r_n$	stany związane z taśmą R	255
A_P	alfabet programu maszyny uniwersalnej Posta	254
A_R	alfabet roboczy maszyny uniwersalnej Posta	254
σP	symbol zapisany w P-kratce	262
σR	symbol zapisany w R-kratce	262
$(\sigma P, p_i)$ $(\sigma P, r_i)$ $(R_l(a_{l_1}, a_{l_2}), p_j)$	sytuacja maszyny	262
$\sigma 0$	symbol zapisany w akumulatorze	264
$!n$	przejście bezwarunkowe	269
$[x]$	całość z x	283
$\mathrm{Nr}(W)$	numer Gödlowski wyrażenia W	292
$C(\Phi)$	numer formuły Φ	293

Skorowidz

Spis rzeczy

PAŃSTWOWE
WYDAWNICTWO NAUKOWE

*

Wydanie I. Nakład 3300+200 egz. Ark. wyd. 18,5. Ark. druk. 19,75. Papier druk. mat. kl. III. 80 g, 61×86. Oddano do składania 23. I. 1969 r. Podpisano do druku 28. I. 1970 r. Druk ukończono w lutym 1970 r. Zam. nr 107/69. J-11. Cena zł 50.—

*

WROCŁAWSKA DRUKARNIA
NAUKOWA