



**UCHWAŁA NR 48  
RADY DYDAKTYCZNEJ DLA KIERUNKÓW INFORMATYKA,  
MATEMATYKA, MACHINE LEARNING  
I INŻYNIERIA OBLICZENIOWA**

z dnia 24 października 2024 r.

**w sprawie zmiany załącznika do *Szczegółowych zasad dyplomowania na kierunku informatyka*, stanowiących załącznik do Uchwały nr 11 z dnia 9 czerwca 2021 r. w sprawie uchwalenia szczegółowych zasad dyplomowania na kierunku informatyka (Dziennik UW RD z 2021r. poz. 196)**

Na podstawie § 68 ust. 2. Statutu Uniwersytetu Warszawskiego (Monitor UW z 2019 r. poz. 190)

w związku z ust. 1 pkt 8 Regulaminu Studiów na Uniwersytecie Warszawskim (Monitor UW z 2019 r. poz. 186) oraz uchwałą nr 4 Uniwersyteckiej Rady ds. Kształcenia w sprawie wytycznych dotyczących procesu dyplomowania na Uniwersytecie Warszawskim (Dziennik UW URK z 2020r. poz. 4), Rada Dydaktyczna postanawia, co następuje:

**§ 1**

Zmienia się załącznik do *Szczegółowych zasad dyplomowania dla kierunku informatyka*, precyzującego zagadnienia na egzamin licencjacki z informatyki, nadając mu treść ujętą w załączniku do uchwały.

**§ 2**

Uchwała wchodzi w życie z dniem podjęcia.

*dr hab. Paweł Goldstein*  
*Przewodniczący Rady Dydaktycznej*



## Zagadnienia na egzamin licencjacki z informatyki

### 1. Proces dyplomowania.

#### Analiza matematyczna

- Co to jest teoria aksjomatyczna? Aksjomatyka liczb rzeczywistych. Aksjomat ciągłości, kresy.
- Zasada indukcji zupełnej. Nierówność Bernoulliego, nierówności między średnimi. Pojęcie zbioru gęstego.
- Granica ciągu: definicja, jednoznaczność, własności arytmetyczne, twierdzenie o trzech ciągach
- Zbieżność ciągów monotonicznych. Podciągi i twierdzenie Bolzano-Weierstrassa
- Twierdzenie Stolza. Funkcja wykładnicza i logarytm (w dziedzinie rzeczywistej).
- Szeregi o wyrazach dodatnich. Warunki konieczne i dostateczne zbieżności, przykłady
- Zbieżność warunkowa i bezwzględna. Kryterium Abela i Dirichleta.
- Mnożenie szeregów. Funkcja wykładnicza w dziedzinie zespolonej.
- Sinus i cosinus w dziedzinie zespolonej. Liczba  $\pi$ .
- Granica funkcji. Granice jednostronne.
- Ciągłość. Własność Darboux i twierdzenie Weierstrassa. Wzmianka o jednostajnej ciągłości.
- Wypukłość i jej różne charakteryzacje. Nierówność Jensena.
- Pochodna: definicja i własności.
- Lemat Fermata, twierdzenie Rolle'a i twierdzenie Lagrange'a o wartości średniej. Wnioski, przykłady.
- Pochodne wyższych rzędów i wzór Taylora.
- Metoda stycznych (gdyby Newton miał komputer...).
- Szeregi potęgowe. Wzór Cauchy'ego-Hadamarda, ciągłość i różniczkowalność sumy szeregu potęgowego, przykłady.
- Funkcja pierwotna. Całka nieoznaczona.
- Całka oznaczona (Newtona, Riemanna) definicja i interpretacja geometryczna. Długość krzywej.
- Różne zastosowania całki oznaczonej.
- Topologia przestrzeni euklidesowej. Norma, metryka, ciągłość funkcji wielu zmiennych rzeczywistych.
- Pochodne cząstkowe i kierunkowe. Różniczka. Interpretacja geometryczna, przykłady.
- Pochodne cząstkowe wyższych rzędów. Wzór Taylora. Warunki dostateczne ekstremów. Przykłady punktów krytycznych.
- Co to jest teoria miary i po co nam ona w ogóle? Przykład Vitaliego,  $\sigma$ -ciała, pojęcie miary zewnętrznej i miary.
- Miara Lebesgue'a: definicje, charakteryzacja, własności. Funkcje mierzalne.
- Teoria całki Lebesgue'a. Ogólna definicja całki. Twierdzenia o zbieżności.
- Całkowanie przez podstawienie. Twierdzenie Fubiniego. Sens geometryczny, przykłady zastosowań.

#### Geometria z algebrą liniową



- Grupy. Ciała. Liczby zespolone, postać trygonometryczna, wzór de Moivre'a, pierwiastki z jednośc, pierwiastki z liczby zespolonej.
- Wielomiany, zasadnicze tw. algebry (bez dowodu).
- Macierze o współczynnikach z ciała. Działania na macierzach.
- Przestrzenie liniowe nad ciałem, podprzestrzeń liniowa, liniowa niezależność, baza, wymiar. Przykłady baz. Część wspólna, suma, suma prosta podprzestrzeni.
- Obraz, jądro i rząd macierzy. Macierze odwracalne.
- Układy równań liniowych. Twierdzenie Kroneckera-Capellego. Opis zbioru rozwiązań. Eliminacja Gaussa.
- Wyznaczniki i ich własności. Wzory Cramera.
- Przekształcenia liniowe i funkcjonały. Macierz przekształcenia liniowego. Rząd, obraz i jądro przekształcenia liniowego oraz macierzy. Izomorfizm przestrzeni liniowych.
- Przestrzeń sprzężona, bazy sprzężone, macierz zmiany bazy, związek z przekształceniami liniowymi.
- Podobieństwo macierzy. Wartość własna, wektor własny, widmo macierzy/przekształcenia liniowego. Wielomian charakterystyczny. Diagonalizacja przekształcenia liniowego/macierzy. Informacja o tw. Jordana.
- Przestrzenie euklidesowe/unitarne. Iloczyn skalarny, norma euklidesowa, pojęcie kąta. Baza ortogonalna/ortonormalna, tożsamość Parsewala. Ortogonalizacja Grama-Schmidta. Dopelnienie ortogonalne i rozkład ortogonalny przestrzeni, rzut ortogonalny. Izometrie, macierze ortogonalne/unitarne.
- Formy hermitowskie i symetryczne. Przystawanie macierzy. Diagonalizacja macierzy symetrycznych/hermitowskich. Kryterium Sylwestera.

## Podstawy matematyki

- Rachunek zdań i jego własności. Wprowadzenie do rachunku kwantyfikatorów.
- Operacje na zbiorach, w tym działania nieskończone.
- Relacje i funkcje oraz ich podstawowe własności.
- Relacja równoważności, zasada abstrakcji.
- Liczby naturalne. Zasada indukcji.
- Równoliczność. Zbiory skończone i nieskończone, przeliczalne i nieprzeliczalne.
- Twierdzenie Cantora i twierdzenie Cantora-Bernsteina.
- Porządki częściowe i liniowe. Kresy. Zastosowania lematu Kuratowskiego - Zorna.
- Porządki dobre i dobrze ufundowane. Indukcja.
- Pojęcie dowodu formalnego. Systemy dowodzenia dla rachunku zdań, twierdzenie o pełności.
- Struktury relacyjne. Język pierwszego rzędu: semantyka, twierdzenie o pełności.

## Wstęp do programowania

- Pojęcie algorytmu:
  - historia powstania pojęcia algorytmu
  - przykłady algorytmów (Euklidesa, Archimedesza, Hornera, rozwiązywanie równań liniowych i kwadratowych)
  - pojęcie dziedziny algorytmicznej



- Podstawy języka programowania:
  - notacja do opisu składni języka (np. gramatyki bezkontekstowe lub BNF),
  - pojęcie składni i semantyki
  - dziedzina algorytmiczna:
    - wyrażenia arytmetyczne (liczby całkowite i zmiennopozycyjne)
    - wyrażenia logiczne
    - znaki i napisy
  - zmienne, ich typ, instrukcja przypisania
  - definiowanie funkcji (bez rekurencji):
    - parametry
    - wartość przekazywana i typ void
    - wywoływanie funkcji
    - zmienne lokalne
  - instrukcja warunkowa,
  - pętla while
  - pętla for
  - instrukcja wyboru
  - znaki i napisy
  - wczytywanie i wypisywanie
  - instrukcja pusta
  - reprezentacja liczb całkowitych
  - reprezentacja liczb zmiennopozycyjnych, pojęcie zakresu i błędów zaokrągleń
- Dekompozycja problemu i weryfikacja programów:
  - specyfikacja problemu, warunek początkowy i końcowy,
  - częściowa poprawność i własność stopu,
  - asercje
  - formuły Hoare'a
  - niezmienniki pętli
  - własność stopu i metody jej dowodzenia
  - uzasadnianie poprawności programów
  - przykłady dekompozycji problemów i weryfikacji programów
- Typy danych:
  - tablice,
  - struktury,
  - unie,
  - deklaracje typów
  - reguły zgodności typów
- Pliki:
  - nośniki pamięci
  - pliki tekstowe
  - mechanizm buforowania
  - standardowe wejście i wyjście jako strumienie
- Przydatne narzędzia i technalia
  - makrodefinicje i pre-processing



- definicje stałych
- klasy: typy obiektów z ograniczonym zestawem metod operujących na nich
- wywoływanie metod obiektu
- metody wywoływane implicite: konstruktory, destruktory, kopiowanie, rzutowanie typów
- przeładowanie operatorów
- typy wyliczeniowe
- przydatne typy danych
- Typy wskaźnikowe:
  - pojęcie wskaźnika, dereferencja,
  - przekazywanie argumentów (i wyników) przez wartość, wskaźnik i referencję
  - zmienne globalne i lokalne,
  - pamięć alokowana dynamicznie,
  - smart pointers -- wskaźniki unikalne i współdzielone
- Modularyzacja i bariery abstrakcji
  - pliki nagłówkowe i implementacja
- Rekurencja:
  - rekurencja i jej sposób implementacji
  - rekurencyjne wyrażanie problemów
  - weryfikacja programów rekurencyjnych
  - analiza złożoności algorytmów:
  - rachunek rzędów funkcji
  - koszty algorytmu: czasowy i pamięciowy, pesymistyczny i średni
  - rozmiar danych
  - analiza programów rekurencyjnych
  - koszt zamortyzowany
  - przykłady wyznaczania kosztów
- Metoda "dziel i rządź":
  - metoda inkrementacyjna
  - podział binarny, w tym wyszukiwanie binarne
  - przykłady -- algorytmy sortowania
- Dynamiczne struktury danych:
  - typy wskaźnikowe
  - wskaźnikowa realizacja list
  - podstawowe operacje na listach
  - listy jednokierunkowe, dwukierunkowe i cykliczne
  - stosy i kolejki
  - atrapy i strażnicy
- Drzewa:
  - drzewa binarne, BST
  - implementacja drzew dowolnego rzędu
  - obiegi drzew
- Przydatne biblioteczne struktury danych:
  - słowniki (mapy)
  - tablice haszujące



- zbiory
- kolejki
- stosy
- Programowanie z nawrotami:
  - przeszukiwanie pełnej przestrzeni stanów
  - heurystyki przyspieszające
  - ucinanie rekursji
- Programowanie dynamiczne
  - metoda spamiętywania
  - tablicowanie
  - programowanie dynamiczne na drzewach
- Programowanie zachłanne:
  - algorytm Huffmana
  - inne przykłady
- Przeszukiwanie grafów:
  - implementacja macierzowa i listowa grafów
  - przeszukiwanie wszerz
  - przeszukiwanie w głąb
  - algorytm Floyda-Warshalla

## Architektura komputerów i systemy operacyjne

- Architektura komputera:
  - architektura a organizacja komputera
  - model von Neumanna
  - architektury Harward i Princeton
  - architektury wspierające zrównoleganie: superskalarna, wektorowa
  - zrównoleganie a dostęp do pamięci, architektury wieloprocessorowe i wielordzeniowe, modele SMP i NUMA
- Mikroprocesor od wewnątrz:
  - rejestry
  - asembler, kod maszynowy, lista rozkazów
  - tryby adresowania
  - CISC a RISC
  - przetwarzanie potokowe
- Mikroprocesor w interakcji ze światem zewnętrznym:
  - system przerwań, wyjątki
  - sprzętowe wsparcie dla systemów operacyjnych
  - poziomy ochrony, wywoływanie usług systemowych, segmentacja i stronicowanie
  - interfejs binarny aplikacji (ABI)
  - obsługa wejścia-wyjścia
  - DMA
- Hierarchia pamięci:
  - pamięć operacyjna



- pamięci podręczne
- pamięci masowe
- macierze dyskowe
- Wprowadzenie do systemów operacyjnych:
  - zadania systemu operacyjnego
  - budowa systemu operacyjnego
  - funkcje systemowe
- Procesy i zarządzanie nimi:
  - pojęcie procesu
  - stany procesu i przejścia między nimi
  - podział czasu, wieloprogramowość
  - synchronizacja na poziomie jądra
  - szeregowanie procesów
- Zarządzanie pamięcią:
  - przestrzeń adresowa procesu
  - strategię przydziału pamięci
  - translacja adresów
  - pamięć wirtualna
- Systemy plików:
  - koncepcja pliku
  - realizacja systemu plików warstwy logicznej i fizycznej
  - przykłady konkretnych implementacji
- System wejścia-wyjścia:
  - urządzenia blokowe i znakowe
  - tablice rozdzielcze
  - pliki specjalne
  - struktura dysku
- Wirtualizacja
- Interpreter poleceń:
  - podstawowe polecenia: diff, grep, find, ps, wc, sort, cut, tail, head itp.
  - potoki, strumienie, przekierowania
  - skrypty bash (instrukcje if, for itp.), argumenty wywołania
  - skrypty do testowania programów
- Podstawy administracji systemem i maszyna wirtualna:
  - maszyna wirtualna
  - zarządzanie użytkownikami
  - prawa dostępu do plików i katalogów
  - instalowanie pakietów oprogramowania
  - pliki konfiguracyjne .bash\_profile, .profile, .bashrc itp.
  - katalogi /etc, /usr, /home itp.
  - logi systemowe
- Język C
  - pliki nagłówkowe, preprocesor, kompilator, linker, program make
  - wskaźniki, sarta, alokacja pamięci, tablica dynamiczna



- dynamiczne struktury danych, BST, wskaźniki do wskaźników, valgrind
- obsługa plików
  - funkcje systemowe: open, close, read, write itp.
  - funkcje z biblioteki standardowej: fopen, fclose, fread, fwrite itp.
  - pliki specjalne: /dev/null, /dev/zero, /dev/random, /dev/urandom itp.
- Programowanie niskopoziomowe
  - asemblacja, dezasemblacja, łączenie
  - podstawowe rozkazy asemblera x86
  - ABI – integracja z programem w C
  - współpraca z systemem operacyjnym
  - debugger

## Matematyka dyskretna

- Indukcja matematyczna i rekurencje
- Sumy skończone
- Współczynniki dwumianowe
- Permutacje i podziały
- Funkcje tworzące i ich zastosowania
- Metody zliczania
  - enumeratory
  - zasada włączania-wyłączania
- Asymptotyka:
  - notacja asymptotyczna (O, Omega, Theta, o, omega)
  - twierdzenie o rekurencji uniwersalnej
- Elementarna teoria liczb:
  - podzielność, liczby pierwsze, rozkład na czynniki pierwsze
  - NWD i algorytm Euklidesa
- Arytmetyka modularna:
  - małe twierdzenie Fermata i twierdzenie Eulera
  - chińskie twierdzenie o resztach
  - rozwiązywanie równań modularnych
- Elementy kryptografii: test Millera-Rabina i system RSA
- Teoria Poly: zastosowanie teorii grup w kombinatoryce
- Grafy:
  - ścieżki, drzewa i cykle
  - cykle Eulera i Hamiltona
  - grafy dwudzielne, skojarzenia i twierdzenie Halla
  - planarność
  - kolorowanie grafów

## Programowanie obiektowe





- 
- Wprowadzenie do obiektowości, pojęcie klasy, UML
  - Składnia Javy (identyfikatory, literały, proste instrukcje, typy ref. i wart., tablice, GC, new, null, ref/wart, funkcje (met.), parametry, dokumentacja)
  - Kapsułkowanie, konstruktory i inicjalizacja
  - Dziedziczenie, klasy abstrakcyjne, interfejsy, polimorfizm
  - Wyjątki nadzorowane i nienadzorowane
  - Typy uogólnione
  - Kolekcje
  - GUI
  - Wzorce projektowe z przykładami (singleton, fabryka, adapter, dekorator, obserwator i inne)
  - Wejście/wyjście
  - Wyrażenia regularne
  - Opcjonalnie: łączenie z funkcyjnością, JVM, obiektowość w innych językach (C++ czy Python)
  - Kompilacja, maszyna wirtualna Javy, pakiety, IDE, dokumentacja
  - Interfejs dla klasy (tablice opakowane w klasę, kolejka/stos, graf)
  - Programowanie imperatywne i tablice, klasy stos/kolejka, listy
  - Narzędzia: git, debugger, ew. inne jak np. SpotBugs
  - Wyrażenia
  - Wyjątki nadzorowane i nienadzorowane
  - Rozdzielanie odpowiedzialności między klasy, luźne sprzężenie/wysoka spójność
  - JUnit
  - Typy uogólnione
  - Kolekcje - używanie istniejących
  - Kolekcje - tworzenie nowych
  - Lambdy oraz java.util.stream
  - Wzorce
  - Narzędzia np. maven i google guava

## Algorytmy i struktury danych

- Podstawowe zasady analizy algorytmów.
- Metody projektowania wydajnych algorytmów.
- Sortowanie.
- Selekcja.
- Kolejki priorytetowe.
- Wyszukiwanie i słowniki.
- Problem „Find-Union” i jego zastosowania.
- Algorytmy grafowe.
- Wyszukiwanie wzorca w tekstach.
- Tekstowe struktury danych.

## Bazy danych

- Co to jest baza danych?



- Modelowanie danych: tabele, drzewa, grafy.
- Języki zapytań: pisanie zapytań, siła wyrazu, ewaluacja.
- Metadane: więzy i zależności.
  
- Języki zapytań dla relacyjnych baz danych.
  - Język deklaratywny: logika pierwszego rzędu (FO).
  - Język imperatywny: algebra relacji (RA).
  - Równoważność języków pomaga pisać zapytania w SQL-u.
  
- Specyfika SQL-a: agregacja i nulle.
  - Zbiory a multizbiory, agregacja, funkcje okienkowe.
  - Różne role wartości NULL, semantyka w SQL-u.
  - Różne złączenia.
  
- Rekurencja.
  - Czego się nie da wyrazić w FO: osiągalność
  - RA z iteracją (while, inflationary while).
  - Rekurencja w SQL-u.
  
- Realizacja zapytań
  - Model I/O: pamięć zewnętrzna, przetwarzanie w blokach, sortowanie, indeksy.
  - Plan logiczny: optymalizacja wyrażenia za pomocą reguł przepisywania i heurystyk (statystyki, SIP).
  - Plan fizyczny: wybór algorytmu realizacji operacji, planowanie kolejności, zrównoleglanie, pipelining.
  
- Pojęciowy model danych.
  - Modelowanie dziedziny jako analiza rzeczywistości.
  - Obiektowe modelowanie dziedziny (UML lub EER).
  - Pułapki modelowania.
  
- Logiczny model danych.
  - Od modelu dziedziny do projektu bazy danych.
  - Więzy: zależności funkcyjne, klucze, zależności inkluzji.
  - Redundancja i anomalie, normalizacja.

## Programowanie współbieżne

- Podstawowe pojęcia współbieżności: proces, przeplot, wykonanie współbieżne.
- Poprawność programów współbieżnych.
- Modele współbieżności.
- Klasyczne problemy współbieżności: wzajemne wykluczanie, czytelnicy i pisarze, pięciu filozofów, producenci i konsumenci.
- Wybrane algorytmy wzajemnego wykluczania niekorzystające ze wsparcia sprzętu ani systemu operacyjnego.
- Definicja i różne semantyki semaforów: semafor Dijkstry, semafony słabe, silne, silnie uczciwe.
- Definicja i różne semantyki monitorów: monitor Hoare'a, realizacja za pomocą biblioteki pthread.



- Metody weryfikacji poprawności programów współbieżnych: LTL, model checking, metody dedukcyjne.
- Metody synchronizacji w modelu rozproszonym: komunikacja synchroniczna i asynchroniczna (przestrzeń krotek).
- Spójność i modele spójności: llinearizability (or atomicity), sequential consistency, causal consistency, eventual consistency.
- Wydajność w modelu współbieżnym: praca, rozpiętość, zrównoleglenie, przyspieszenie.
- Wybrane algorytmy rozproszone (wzajemne wykluczanie w systemach rozproszonych, synchronizacja zegarów logicznych, uzgadnianie).
- Procesy i wątki oraz metody ich synchronizacji w systemach zgodnych ze standardem POSIX.
- Współbieżność w Javie (java.util.concurrent, implementacja monitora).
- Algorytmy wzajemnego wykluczania bez wsparcia sprzętu i systemu operacyjnego.
- Semafor.
- Monitory.
- Synchroniczna wymiana komunikatów.
- Asynchroniczna wymiana komunikatów.
- Miary wydajności programów współbieżnych.

## Rachunek prawdopodobieństwa

- Przestrzeń probabilistyczna: aksjomaty, własności, schemat klasyczny prawdopodobieństwo geometryczne, miara.
- Prawdopodobieństwo warunkowe: prawdopodobieństwo całkowite, wzór Bayesa, niezależność zdarzeń.
- Dyskretne zmienne losowe: definicja, własności, podstawowe rozkłady – dwupunktowy, dwumianowy, Poissona, geometryczny.
- Parametry rozkładu: wartość oczekiwana, wariancja, momenty, funkcje tworzące prawdopodobieństwa, ich własności oraz zastosowania do wyznaczania parametrów rozkładu.
- Szacowanie ogonów: nierówności Markowa, Czebyszewa, Chernoffa, prawa wielkich liczb.
- Ciągłe zmienne losowe: definicja, własności, rozkład jednostajny, wykładniczy oraz normalny, funkcje charakterystyczne, centralne twierdzenie graniczne.
- Łańcuchy Markowa: definicja oraz podstawowe własności, prawdopodobieństwa oraz średnie czasy dotarcia, klasyfikacja stanów, łańcuchy Markowa o nieskończenie wielu stanach, twierdzenie ergodyczne, zastosowania.



- Algorytmy związane z rachunkiem prawdopodobieństwa: metoda Monte Carlo, próbkowanie z rozkładu.

## Aplikacje WWW

- Wprowadzenie
- CSS
- JavaScript / TypeScript
- DOM, zdarzenia
- Wprowadzenie do Node.js i Express.js
- Przesyłanie danych pomiędzy klientem a serwerem
- Przechowywanie danych z użyciem relacyjnych baz danych
- Identyfikacja i autoryzacja użytkowników, bezpieczeństwo
- Aplikacje jednostronicowe i PWA
- Współczesne frameworki do tworzenia aplikacji
- Grafika, wizualizacje, multimedia
- Budowanie i dostarczanie aplikacji
- Usprawnianie aplikacji
- Użycie mechanizmów aplikacji WWW do tworzenia programów
- WebAssembly

## Inżynieria oprogramowania

- Projekt informatyczny
- Architektura
- Funkcjonalność
- Wzorce
- Specyfikacja
- Jakość
- Planowanie
- Zarządzanie konfiguracją
- Metodyki
- Praca w grupie

## Języki, automaty i obliczenia

- Elementy teorii języków formalnych: słowa, języki, wyrażenia regularne.
- Automaty skończone i twierdzenie Kleene'ego o efektywnej odpowiedniości automatów i wyrażeń.
- Konstrukcje optymalizujące automaty - determinizacja, minimalizacja.
- Języki bezkontekstowe: gramatyki i ich postaci normalne.
- Odpowiedniość gramatyk bezkontekstowych i niedeterministycznych automatów ze stosem.



- Kryteria rozróżniania języków regularnych i bezkontekstowych - lematy o pompowaniu.
- Zagadnienia algorytmiczne: problem niepustości dla automatów i gramatyk, rozpoznawanie języków bezkontekstowych.
- Przykłady zastosowań automatów i gramatyk.
- Uniwersalne modele obliczeń: maszyna Turinga i jej warianty.
- Granice obliczalności: nierozstrzygalność problemu stopu, przykłady praktycznych problemów nierozstrzygalnych.
- Podsumowanie - klasyfikacja gramatyk, modeli obliczeń i języków według hierarchii Chomsky'ego.
- Wprowadzenie do zagadnień złożoności obliczeniowej: klasy P i NP.
- Twierdzenie Cooka-Levina o NP-zupełności SAT.
- Hipoteza  $P \neq NP$  i jej praktyczne implikacje, informacja o pozytywnych zastosowaniach problemów trudnych obliczeniowo, np. w kryptografii.

## Sieci komputerowe

- Sieci komputerowe - podstawowe pojęcia, modele warstwowe
- Warstwa fizyczna i warstwa łącza
  - Charakterystyki różnych sposobów transmisji: światłowód, skrętka miedziana, fale radiowe
  - Ethernet, techniki przełączania, budowa sieci lokalnych, protokół STP, VLAN
  - sieci bezprzewodowe (WiFi, Bluetooth)
- Warstwa sieciowa i transportowa
  - Adresy IP, maska podsieci, adresowanie, przekazywanie pakietów
  - Translacja adresów (NAT), rozgłaszanie, komunikacja grupowa
  - Zawodna komunikacja pakietowa (IP, UDP)
  - Niezawodna komunikacja strumieniowa (TCP)
  - Protokoły wspierające (ARP, ICMP)
  - Zaawansowane trasowanie międzyoperatorskie
- Warstwy wyższe
  - Architektura klient-serwer i P2P
  - System nazw domenowych (DNS)
  - Protokół automatycznej konfiguracji interfejsu sieciowego (DHCP)
- Podstawy bezpieczeństwa
  - Rodzaje ataków sieciowych i sposoby obrony
  - Bezpieczeństwo operacyjne (zapory ogniowe)
  - Szyfrowanie symetryczne i asymetryczne, podpis elektroniczny, ustalanie klucza
  - Bezpieczne protokoły różnych warstw, np. TLS, IPsec

## Wstęp do uczenia maszynowego

- Estymacja parametrów
- Testowanie hipotez
- p-wartości i testowanie wielu hipotez



- Uczenie statystyczne
- Regresja liniowa
- Klasyfikacja
- Metody repróbkiwania, wybór modelu
- Regularyzacja
- Metody drzewiaste
- Maszyny wektorów wspierających
- Podstawy sieci neuronowych
- Redukcja wymiaru
- Klastrowanie