

Zadanie zaliczeniowe nr 1  
(MSUI-PWiO)

Napisz w Smalltalku hierarchię klas opisujących żołnierzy. Wyraż w tej hierarchii następujące fakty:

- wszyscy żołnierze są ludźmi,
- każdy szeregowiec, sierżant, porucznik i kapitan jest żołnierzem,
- oddziały wojskowe dzielą się na kompanie i plutony.

Stwórz też klasę opisującą kompanię. Tak zaprojektuj klasy by:

- każdy człowiek znał swoje imię i nazwisko,
- każdy żołnierz znał swoją kompanię,
- każdy kapitan znał swoich bezpośrednich podwładnych (poruczników),
- każdy porucznik znał swojego przełożonego (kapitana) i swoich podwładnych (szeregowców),
- każdy żołnierz znał swojego przełożonego (dla kapitana wpisujemy tu nil),
- każda kompania składała się z plutonów,
- każda kompania znała swoje plutony i swojego dowódcę (kapitana) oraz zaopatrzeniowca (sierżanta),
- każdy pluton znał swoich żołnierzy (dowódcę – porucznika i szeregowców),
- każdy pluton znał swoją kompanię,
- każdy oddział wojskowy miał nazwę.

Każdy z ludzi musi umieć podać swój opis w postaci napisu (komunikat `printString`). Opis ma zawierać wszystkie informacje, które zna dany człowiek (np. nazwę oddziału, czy nazwisko i stopień przełożonego – o ile ma przełożonego). Kompania powinna wypisywać swój opis do okienka Transcript. Opis kompanii to jej nazwa opis dowódcy i zaopatrzeniowca oraz opisy poszczególnych plutonów. Opis plutonu oprócz nazwy swojej i nazwy swojej kompanii powinien zawierać spis żołnierzy plutonu. Pamiętaj, żeby nie pisać od nowa rzeczy, które raz już napisałeś i pamiętaj o zasadzie „leniwy ale uczciwy”.

Napisz fragment programu w Smalltalku, który utworzy kilka (ok. 10) obiektów reprezentujących różnych żołnierzy i różne oddziały, umieści wszystkich w jednej kompanii, a następnie wypisze jej opis. Zadbaj o to, by tworzenie nowych obiektów było jak najprostsze.

- Pisanie do okienka Transcript realizuje następujące wyrażenie: `Transcript show: <napis>`. Przejście do nowego wiersza daje metoda `cr` (okienka Transcript). Napisy można ze sobą łączyć operacją `,`.

Posługiwanie się kolekcją:

- `OrderedCollection new`, daje nową (pustą) kolekcję,
- `k add: x`, dodaje na koniec kolekcji `k` nowy element `x`,
- `k do: [ :p | <zrób coś z p> ]`, wykonuje czynność `<zrób coś z p>` dla każdego elementu kolekcji `k`. Opis czynności to dowolny ciąg instrukcji smalltalkowych porozdzielanych kropkami. Można w nim odwoływać się do obiektu `p` (nie podając dwukropka). Uwaga: zarówno nawiasy `[ ]` jak i kreska pionowa `|` są w tym przykładzie potrzebne.