

# PODSTAWY PROGRAMOWANIA DEKLARATYWNEGO

## HASKELL

### Ćwiczenia 6 (20p.)

#### Zadanie 1. (3p.)

Napisać moduł o nazwie *Nazwisko* (*Nazwisko* oznacza nazwisko Studenta) z funkcjami *podzbior*, *iloczyn*, *suma*, *roznica* (podzbiór, iloczyn zbiorów, suma zbiorów, różnica zbiorów odpowiednio).

(Funkcje *iloczyn* i *suma* zbiorów napisane na ćwiczeniach 3. Przy definiowaniu funkcji *podzbior* i *roznica* nie używać funkcji wbudowanych Haskella odpowiadających tym funkcjom).

Napisać program (osobny plik hs), w którym można używać funkcji zdefiniowanych w module *Nazwisko*.

#### Zadanie 2. (2p.)

Dla danej listy *l* wypisać listę par [(element, liczba wystąpień), ...].

Przykład: w [2,3,4,1,2,5,3,2,4,4,2] = [(2,4),(3,2),(4,3),(1,1),(5,1)]

w ['a','a','b','a'] = [('a',3),('b',1)]

#### Zadanie 3. (3p.)

Napisać funkcję, której wartością jest długość najkrótszej gałęzi w drzewie binarnym.

#### Zadanie 4. (3p.)

Napisać funkcję, której wartością jest *True*, gdy drzewo binarne *t* ma w węzłach liczby całkowite i jest uporządkowane, tzn. elementy w lewym poddrzewie są mniejsze od korzenia, a w prawym poddrzewie – większe.

#### Zadanie 5. (5p.)

Napisać program do przekształcania dowolnego drzewa binarnego do postaci drzewa uporządkowanego.

#### Zadanie 6. (4 p.)

Zdefiniować typ *Roman* następująco: `data Roman = Roman Integer`

i zdefiniować instancje klasy *Show* i *Num* dla tego typu. Funkcja *show* powinna wyświetlać liczby w postaci liczb rzymskich, np.

`show (Roman 99) = "IC"`

`show (Roman 1327) = "MCCCXXVII"`.

Instancja klasy *Num* powinna spełniać:

$(\text{Roman } n) + (\text{Roman } m) \sim (\text{Roman } n+m)$

**Uwaga:**

Wykonane zadania należy przekazać do **07.04.2019, 23:55** przez OLAT „Ćwiczenia 6-pn”.  
Wszystkie definicje funkcji z zadań 2-6 mają być zapisane w jednym pliku .hs z numerem zadania w komentarzu. Nazwa pliku ma zawierać nazwisko Studenta i numer ćwiczeń.