

PODSTAWY PROGRAMOWANIA DEKLARATYWNEGO

HASKELL

Ćwiczenia 8

Zadania powtórkowe

Zadanie 1.

Podać typ danej funkcji. Obowiązuje znajomość wszystkich funkcji omówionych na wykładzie.
Np. typ funkcji map:

`map :: (a -> b) -> [a] -> [b]`

Podać wartości danej funkcji. Np.

Wartością `map (>1) [3,2,1,0,-1]` jest `[True,True,False,False,False]`

Wartością `map (*2) [3,2,1,0,-1]` jest `[6,4,2,0,-2]`

Zadanie 2.

Napisać definicję funkcji obliczającej n-ty wyraz ciągu danego wzorem $a_n = a_{n-1} + 5a_{n-2}$ ($a_1=0$, $a_2=3$) a) rekurencyjną, b) w wersji „akumulatorowej”.

Zadanie 3.

Napisać definicję funkcji, która w liście przestawia a) pierwszy element z drugim, b) pierwszy element z ostatnim, c) drugi element z przedostatnim.

Zadanie 4.

Napisać definicję funkcji, której wartością jest *True*, jeśli liczba wystąpień różnych elementów liście *l* jest taka sama.

Przykład: `f [1,2,3,2,4,1,3] = False`

`f [1,2,3,2,3,1] = True`

Zadanie 5.

Napisać funkcję sprawdzającą równość dwóch list.

Zadanie 6.

Napisać funkcję sprawdzającą równość dwóch zbiorów.

Zadanie 7.

Napisać funkcję, która porządkuje niemalejąco elementy listy liczbowej wykorzystując algorytm sortowania a) przez proste wstawianie, b) „bąbelkowego”.

Zadanie 8.

Napisać funkcję, która dla dwóch uporządkowanych niemalejąco list liczbowych $l1$ i $l2$ daje w wyniku uporządkowaną niemalejąco listę elementów z list $l1$ i $l2$.

Zadanie 9.

Poniższa funkcja ma sprawdzać, czy podana lista jest pusta. Dlaczego definicja ta jest niepoprawna? Jak należy poprawić?

```
pustal :: [a] -> Bool
pustal x = (x == [])
```

Zadanie 10.

Poniższy program ma konwertować ciągi cyfr (String) na liczby (Int). Wyjaśnić, na czym polega błąd w definicji i ten błąd usunąć.

```
import Data.Char
naInt :: String -> Int
naInt = foldl (\x acc -> acc * 10 + digitToInt x) 0
```

Zadanie 11.

Dla danej listy par (x,y) należy zwrócić listę tych par posortowaną wg odległości od punktu $(0, 0)$.

Przykład: $f [(0,3),(1,1),(2,1),(1,0)] = [(1,0),(1,1),(2,1),(0,3)]$