

## Ćwiczenie 3

### Haskell - definiowanie funkcji i operatorów infiksowych

## 1 Wprowadzenie

### 1.1 Definiowanie funkcji przy użyciu wzorców

```
nazwaFunkcji wzorzec_1 = wyrażenie_1
nazwaFunkcji wzorzec_2 = wyrażenie_2
...
nazwaFunkcji wzorzec_n = wyrażenie_n
```

UWAGA: Kolejność dopasowywania wzorców ma znaczenie.

Przykłady:

```
f1 0 = 1
f1 1 = 3
f1 2 = 10
f1 _ = 0

f2 0 = 1
f2 x = 2*x
```

UWAGA: Wzorzec `_` to tzw. "dzika karta" (*wild card*).

### 1.2 Definiowanie funkcji przy użyciu strażników

```
nazwaFunkcji argumenty_wejściowe | warunek_1 = wyrażenie_1
  | warunek_2 = wyrażenie_2
  ...

nazwaFunkcji argumenty_wejściowe | warunek_1 = wyrażenie_1
  | warunek_2 = wyrażenie_2
  ...
  | otherwise = wyrażenie_n
```

UWAGA: Kolejność strażników nie ma znaczenia.

### 1.3 Krotki

Krotki są elementami zbiorów będących iloczynami kartezyjańskimi zbiorów. Zapis krotki:

```
(element_0, element_1, ...)
```

Elementy krotki mogą być różnych typów.

## 1.4 Definiowanie operatorów infiksowych

```
reguła_łączności priorytet symbol
argument_1 symbol argument_2 = wyrażenie
```

Przykładowe reguły łączności:

- infixl - łączność lewostronna.
- infixr - łączność prawostronna.

Priorytet operatora jest liczbą z zakresu od 0 (najniższy priorytet) do 9 (najwyższy priorytet).

Przykład:

```
infixl 8 +++
x +++ y = x+2*y
```

Można dodać informację o typach, np.:

```
infixl 8 +++
(+++) :: Integer -> Integer -> Integer
x +++ y = x+2*y
```

## 2 Zadania

### 2.1

Zdefiniuj moduł z funkcją *signum*, która dla argumentu rzeczywistego  $x$  określona jest wzorem:

$$\text{signum}(x) = \begin{cases} 1 & \text{dla } x > 0 \\ 0 & \text{dla } x = 0 \\ -1 & \text{dla } x < 0 \end{cases}$$

Przetestuj działanie funkcji.

### 2.2

Zdefiniuj moduł z funkcjami, które dla argumentu  $x$  całkowitego nieujemnego określone są wzorami:

- $a(x) = \begin{cases} 0 & \text{dla } x < 2 \\ 5 & \text{dla } x \geq 2 \end{cases}$
- $b(x) = \begin{cases} -1 & \text{dla } x < 3 \\ 0 & \text{dla } x \geq 3 \text{ i } x \leq 5 \\ 1 & \text{dla } x > 5 \end{cases}$

UWAGA: Wykorzystaj definiowanie funkcji przy użyciu wzorców. Przetestuj działanie funkcji.

### 2.3

Zdefiniuj moduł z funkcjami, które dla argumentu rzeczywistego  $x$  określone są wzorami:

$$\bullet a(x) = \begin{cases} \sin(x) & \text{dla } x > 0 \\ x & \text{dla } x = 0 \\ \cos(x) & \text{dla } x < 0 \end{cases}$$
$$\bullet b(x) = \begin{cases} \sqrt{x+1} & \text{dla } x > 0 \\ 0 & \text{dla } x = 0 \\ \sqrt{|x-1|} & \text{dla } x < 0 \end{cases}$$

Przetestuj działanie funkcji.

### 2.4

Zdefiniuj moduł z funkcjami obliczającymi silnię (z liczby całkowitej nieujemnej) oraz współczynnik dwumianowy Newtona (UWAGA: funkcja obliczająca współczynnik dwumianowy Newtona powinna wykorzystywać funkcję obliczającą silnię). Przetestuj działanie funkcji.

### 2.5

Zdefiniuj moduł z funkcjami rekurencyjnymi wyznaczającymi  $n$ -te wyrazy ciągów:

$$\bullet a(x) = \begin{cases} 2 & \text{dla } x = 0 \\ a(x-1) + 3x & \text{dla } x > 0 \end{cases}$$
$$\bullet b(x) = \begin{cases} 1 & \text{dla } x = 0 \\ 3 & \text{dla } x = 1 \\ 2b(x-1) + 3b(x-2) & \text{dla } x > 1 \end{cases}$$

Przetestuj działanie funkcji.

### 2.6

Zdefiniuj moduł zawierający infiksowe operatory arytmetyczne dla liczb zespolonych oraz funkcje zwracające część rzeczywistą oraz część urojoną liczby zespolonej. Przyjmij, że liczba zespolona jest zapisywana jako krotka dwóch liczb rzeczywistych. Przetestuj działanie operatorów i funkcji.

### 2.7

Zdefiniuj moduł zawierający infiksowe operatory dodawania, odejmowania, mnożenia skalarnego, mnożenia wektorowego wektorów w przestrzeni  $\mathbb{R}^3$ . Przyjmij, że wektor jest zapisywany jako krotka trzech liczb rzeczywistych. Przetestuj działanie operatorów.