Programowanie funkcyjne - laboratoria

Wojciech Typer

zadanie 1 power x y = power y^x

p2 = power 4 $\rightarrow$ power 4 y = y^4
p3 = power 3
(p2 . p3) 2 = p2(p32) = p 2 8 = 8^4 = 4096
p2 :: Int -> Int
p3 :: Int -> Int
(p2 . p3) :: Int -> Int
Wyrażenia lambda:
power = $/x \rightarrow /y \rightarrow y^x$
p2 = $/y \rightarrow y^4$
p3 = $/y \rightarrow y^3$

zadanie 4

plus = λ xy -> x + y
multi = λ xy -> x * y

zadanie 5

haskell:
$\lambda x \rightarrow 1 + x * (x + 1)$
python:
f = lambda x: 1 + x * (x + 1)

zadanie 6

Ustalmy zbiory A, B, C. Niech

$$\text{curry} : C^{B\times A} \rightarrow (C^B)^A$$

będzie funkcją zadaną wzorem:

$$\text{curry}(\varphi) = \lambda a \in A \rightarrow (\lambda b \in B \rightarrow \varphi(b, a)).$$

oraz niech

$$\text{uncurry} : (C^B)^A \to C^{B\times A}$$

będzie zadana wzorem:

$$\text{uncurry}(\psi)(b,a) = (\psi(a))(b).$$

1. Pokaż, że $\text{curry} \circ \text{uncurry} = \text{id}_{(C^B)^A}$ oraz $\text{uncurry} \circ \text{curry} = \text{id}_{C^{B\times A}}$.
2. Wywnioskuj z tego, że $|(C^B)^A| = |C^{B\times A}|$. Przypomnij sobie dowód tego twierdzenia, który poznałeś na pierwszym semestrze studiów.
3. Spróbuj zdefiniować w języku Haskell odpowiedniki funkcji `curry` i `uncurry`.

---

1. Pokażemy, że $\text{curry} \circ \text{uncurry} = \text{id}_{(C^B)^A}$ oraz $\text{uncurry} \circ \text{curry} = \text{id}_{C^{B\times A}}$.
   - $\text{curry} \circ \text{uncurry}$
   $$\begin{aligned}(\text{curry} \circ \text{uncurry})(\psi) &= \text{curry}(\text{uncurry}(\psi)) \\ &= \text{curry}(\lambda a \in A \to (\lambda b \in B \to \psi(a)(b))) \\ &= \lambda a \in A \to (\lambda b \in B \to \psi(a)(b)).\end{aligned} \tag{1}$$
   - $\text{uncurry} \circ \text{curry}$
   $$\begin{aligned}(\text{uncurry} \circ \text{curry})(\varphi) &= \text{uncurry}(\text{curry}(\varphi)) \\ &= \text{uncurry}(\lambda a \in A \to (\lambda b \in B \to \varphi(b,a))) \\ &= \lambda b \in B \to (\lambda a \in A \to \varphi(b,a)).\end{aligned} \tag{2}$$

   Z powyższych równań wynika, że $\text{curry} \circ \text{uncurry} = \text{id}_{(C^B)^A}$ oraz $\text{uncurry} \circ \text{curry} = \text{id}_{C^{B\times A}}$. $\square$

2. Możemy pokazać że `curry` i `uncurry` są iniekcjami niewprost, nakładając odpowiednio przeciwne funkcje na obie strony równości:
   - Załóżmy, że $\text{curry}(\varphi_1) = \text{curry}(\varphi_2)$. Wtedy:
   $$\begin{aligned}\text{curry}(\varphi_1)(a)(b) &= \text{curry}(\varphi_2)(a)(b) \\ \varphi_1(b,a) &= \varphi_2(b,a) \\ \varphi_1 &= \varphi_2.\end{aligned} \tag{3}$$
   - Załóżmy, że $\text{uncurry}(\psi_1) = \text{uncurry}(\psi_2)$. Wtedy:
   $$\begin{aligned}\text{uncurry}(\psi_1)(b,a) &= \text{uncurry}(\psi_2)(b,a) \\ \psi_1(a)(b) &= \psi_2(a)(b) \\ \psi_1 &= \psi_2.\end{aligned} \tag{4}$$

A więc istnieje biekcja między $(C^B)^A$ i $C^{B\times A}$, co oznacza, że te zbiory mają taką samą moc. $\square$

3. W języku Haskell funkcje `curry` i `uncurry` można zdefiniować następująco:

```
curry :: ((b, a) -> c) -> a -> b -> c
curry f x y = f (y, x)
```

```
uncurry :: (a -> b -> c) -> (a, b) -> c
uncurry f (x, y) = f x y
```