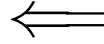


Písomná skúška FP: 15. mája 2023

Táto písomná skúška trvá 90 minút. Počas skúšky je povolené používať laptop s inštalovaným kompilátorom jazyka Haskell `ghc`, resp. `ghci`, `cabal` a/alebo `stack`, `Visual Studio Code` a/alebo `emacs` a/alebo `IntelliJ IDEA`. Iné používateľské aplikácie, najmä komunikačné aplikácie, webové prehliadače a aplikácie pre synchronizáciu súborov, nie sú povolené. Knihy, dokumentácia v papierovej podobe, písané poznámky atď. nie sú povolené.

Túto písomnú skúšku je potrebné vypracovať samostatne, teda bez pomoci niekoho iného a bez komunikácie s niekým iným. Odhalené podvádžanie pri skúške, napr. kopírovanie odpovedí a riešení (aj ich častí), snaha odovzdať cudzie odpovede a riešenia a pod., môže byť penalizované vylúčením študenta z predmetu, nepridelením a/alebo zrušením bodov, prípadne aj disciplinárnym konaním v zmysle Študijného poriadku, ktoré môže viesť k vylúčeniu zo štúdia.

Meno a priezvisko:



1. Naimplementujte funkciu `zip` pomocou `zipWith`. (5 b.)

```
1 zip :: [a] -> [b] -> [(a, b)]
```

Riešenie:

2. Prečo je v jazyku Haskell možné definovať funkciu `bar` takýmto spôsobom? Aký význam má funkcia `curry`? (5 b.)

```
1 foo a b = a + b
2 bar = foo 1
3
4 curry :: ((a, b) -> c) -> a -> b -> c
```

Riešenie:

3. Prečo prvá možnosť skompiluje a vráti hodnotu, a druhá neskompiluje? (5 b.)

```
1 take 0 undefined
2 take undefined []
```

Riešenie:

4. Naimplementujte funkciu `length` pomocou `foldl`/`foldr`. Funkcia `length` vráti dĺžku zoznamu. (5 b.)

```
1 length :: Foldable t => t a -> Int
```

Riešenie:

5. Ako môžeme naimplementovať funkciu `f` tak, aby `foldl` vrátil `Just 3`? (5 b.)

```
1 foldl (+) (0) [1, 2] -- 3
2 foldl f (Just 0) [Just 1, Just 2] -- Just 3
```

Riešenie:

6. Naimplementujte funkciu `foo`, ktorá zo zoznamu `str` vyberie prvky, ktoré majú dĺžku menšiu ako `n`. (5 b.)

```
1 str = [x++[a] | x <- "" : str, a <- ['a'..'z']]
```

Riešenie:

```
1 foo :: Int -> [[Char]]
```

7. Aký je principiálny problém s takýmto zápisom funkcie main? Prečo nemôžeme vrátiť premennú a? Prečo vytlačiť premennú a nie je problém? (5 b.)

```
1 main :: String
2 main = do
3     a <- getLine
4     print a
5     return a
```

Riešenie:

8. Aký je rozdiel medzi funkciami liftA2 a liftM2? Prečo ich potrebujeme? (5 b.)

```
1 liftA2 :: Applicative f => (a -> b -> c) -> f a -> f b -> f c
2 liftM2 :: Monad m => (a1 -> a2 -> r) -> m a1 -> m a2 -> m r
```

Riešenie:

9. Naimplementujte funkciu foo, ktorá zo zoznamu odstráni duplikáty. Pokryte všetky možnosti vstupných parametrov. Prečo je potrebná typeclass Ord? Je potrebná aj typeclass Eq? Prečo je alebo nie je potrebná typeclass Eq? Funkciu foo implementujte pomocou rekurzie. (10 b.)

Riešenie:

```
1 foo :: (Ord a) => [a] -> [a]
```