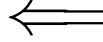


# Písomná skúška FP: 17. mája 2024

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. Overte pravdivosť Collatzovho predpokladu pre  $n \in [1, 1000]$ . Collatzova funkcia je definovaná tak, že pre ľubovoľné  $n$ , ak je  $n$  párne, tak bude stávajúca hodnota predelená 2, v opačnom prípade bude hodnota vynásobená 3 a k tomuto výsledku bude pripočítaná 1, a tento postup je rekurzívne volaný. Podľa Collatzovho predpokladu, pre ľubovoľné počiatočné  $n$ , výpočet skončí hodnotou 1. (5 b.)

1 `collatz :: Int -> Int`

Riešenie:

---

2. Implementuj funkciu, ktorá pre ľubovoľné prirodzené číslo vráti počet volaní Collatzovej funkcie pokým hodnota nie je 1. (5 b.)

1 `collatzN :: Int -> Int`

Riešenie:

3. Naimplementuj funkciu *filter* pomocou *foldr*. Funkcia *filter* vráti zoznam prvkov, ktoré spĺňajú vlastnosť *p*. (5 b.)

1 **filter** :: (a -> Bool) -> [a] -> [a]

Riešenie:

---

4. Pomocou typovej signatúry vysvetli, čo implementuje funkcia *liftM2* (Control.Monad). (5 b.)

1 **liftM2** :: Monad m => (a1 -> a2 -> r) -> m a1 -> m a2 -> m r

Vysvetli m, a1, a2, r, m a1, m a2, m r, (a1 -> a2 -> r), m a1 -> m a2 -> m r.

Riešenie:

5. Naimplementuj funkciu, ktorá zo zoznamu celých čísel odstráni duplikáty. Požiadavka: použi funkciu `group` (`Data.List`), riešenie zapíš pomocou kompozície niekoľkých funkcií vyššieho rádu. Hint: riešenie je na 1 riadok. (5 b.)

1 `dup :: [Int] -> [Int]`

Riešenie:

---

6. Prepíš `do` blok pomocou lambda funkcií. (5 b.)

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

Riešenie:

7. Prečo v jazyku Haskell potrebujeme bind operátor `>>=`? Aké má vlastnosti? (5 b.)

Riešenie:

8. Uved' príklad, kedy je to potrebné, aby typová trieda *Monad* bola asociatívna. (5 b.)

Riešenie:

---

9. Vypočítaj počet výskytu každého znaku v reťazci pomocou *foldr*. Napr. pre "hello world" bude výstup: [( ' ',1),( 'd',1),( 'e',1),( 'h',1),( 'l',3),( 'o',2),( 'r',1),( 'w',1)]. Hint: použite *Data.Map* a funkciu *insertWith*, prázdnu *Map* vytvoríte pomocou *Map.empty*. (10 b.)

```
insertWith :: Ord k => (a -> a -> a) -> k -> a -> Map k a -> Map k a
```

Riešenie:

```
1 import qualified Data.Map as Map
2 countOccurrences :: Ord a => [a] -> Map.Map a Int
```