

Csoportok

Eseményvezérelt alkalmazások

IP-18bEVALKEG | 12

Eseményvezérelt alkalmazások

IP-18bEVALKEG | 91

Eseményvezérelt alkalmazások

IP-18bEVALKEG | 92

8. házi feladat

Kategória:	Házi feladatok
Elérhető:	2022. 11. 03. 23:59
Pótolható határidő:	2022. 11. 11. 23:59
Végső határidő:	2022. 11. 13. 23:59
Kiírta:	Erdei Zsófia

Leírás:

Házi feladat

A házi feladatot egy Homework8 nevű modulként kell beadni. Minden definiálandó függvényhez adjuk meg a hozzá tartozó típuszignatúrát is!

Fontos: a feltöltött tömörített zip állományban egyetlen .hs kiterjesztésű fájl szerepeljen (*hazi.txt*, *Homework.hs.txt* nem felel meg)!

Tipp: Ha szükség van rá, a feladatokat bontsd fel részfeladatokra és definiáljatok segédfüggvényeket!

A bónusz feladat megoldása nem kötelező (és azt nem teszteli az automatikus tesztelő).

DNS

Definiáljuk a **Base** adattípust, amely egy DNS szál egy bázisát reprezentálja és a következő négy paraméter nélküli adatkonstruktossal rendelkezik: **A** , **T** , **G** , **C** . Kérjük az **Eq** és a **Show** típusosztályok automatikus példányosítását!

Komplementer

Egy DNS szál bázisokból áll (adenin - A, guanin - G, citozin - C, timin - T), minden bázisnak van egy párja (A-T, C-G). Definiáld az **isComplement :: [Base] -> [Base] -> Bool** függvényt, amely eldönti, hogy a két paraméterként megadott DNS szál egymás komplementere-e! Egy DNS szál komplementerét úgy állíthatjuk elő, hogy a bázisokat kicseréljük a megfelelő párjukra. Feltételezhetjük, hogy a két input lista hossza megegyezik.

```
isComplement [] [] == True
isComplement [A,T,C,G] [T,A,G,C] == True
isComplement [A,T,A,T,C,G] [T,A,T,A,G,C] == True
isComplement [A,T,A,T,C,G] [T,A,T,A,G,G] == False
isComplement [A,T,A,T,C,G] [A,A,T,A,G,C] == False
```

Tranzakciók

Definiáljuk a **Transaction** adatszerkezetet ami egy bankszámlán történő egyszerű tranzakciókat reprzentálja. Három konstruktora legyen, egy az másnak való átutalásnak (**Transfer**), egy a vásárlásnak (**Purchase**) és egy a nekünk való átutalásnak (**Receive**). Minden konstruktornak legyen egy **Int** típusú pparamétere ami a tranzakció folyamán átutalt pénzt reprezentálja. A **Transfer** és **Receive** konstruktoroknak legyen még egy **Int** típusú paramétere ami a másik bankszámlának a számát reprezentálja. A **Receive** konstruktornak legyen egy **String** típusú paramétere amely az utaláshoz csatolt üzenetet reprezentálja! Kérjük a **Show** típusosztály automatikus példányosítását!

Pénzösszeg

Definiáljuk a **netGain :: [Transaction] -> Int** függvényt amely megadja hogy az adott utalások során, mennyi pénz szereztünk/vesztettünk!

```
netGain [Purchase 1000] == -1000
netGain [Purchase 100, Transfer 500 123, Receive 700 456 "Fizetés"] == 100
netGain [Receive 15000 123 "Ebédpénz", Purchase 10000, Purchase 5000] == 0
```

Negatívba ment-e

Definiáljuk a `wasNegative :: [Transaction] -> Bool` függvényt amely eldönti hogy az adott utalások során voltunk-e valamikor negatívban!

```
wasNegative [Purchase 1000]
wasNegative [Purchase 100, Transfer 500 123, Receive 700 456 "Fizetés"]
not $ wasNegative [Receive 15000 123 "Ebédpénz", Purchase 10000, Purchase 5000]
wasNegative [Receive 15000 123 "Ebédpénz", Purchase 15001, Receive 2 456 "Visszajárás"]
```

Gyakorló feladat mintaillesztéshez

Ilyen elméleti feladat várható ZH-ban!

Adott a következő `foo1` függvény:

```
foo1 :: ([[a]],([a],String)) -> Int
foo1 ([_],(a:b,"Hello")) = 0
foo1 ([_],(a:b,'H':'e':'l':'l':'o':xs)) = 1
foo1 ([_,_,_],(a:b,'H':'e':'l':'l':'o':_:_:[])) = 2
```

A függvény a `foo1Solution = ([[1,2]],([3],"Hello world!"))` bemenetre az `1` eredményt adja.

```
foo1 foo1Solution1 == 1
```

1. Az előző példa alapján adj meg olyan konstans függvényt `foo1Solution2` néven, amelyet ha paraméterül adunk a `foo1` függvénynek 2 lesz a visszatérési értéke!

```
foo1 foo1Solution2 == 2
```

Adott a következő adattípus:

```
data Gyumolcs = Alma Int Char | Barack | Cseresznye String
```

1. Adj meg olyan konstans függvényt `foo2Solution2` néven, amelyet ha paraméterül adunk a `foo2` függvénynek 2 lesz a visszatérési értéke!

```
foo2 :: (Gyumolcs, String) -> Int
foo2 (Alma 12 'a', "Piros") = 0
foo2 (Barack, 's':_a':xs) = 1
foo2 (Cseresznye ('p':xs), 'p':[]) = 2
```

```
foo2 foo2Solution2 == 2
```

Megoldás

Letöltés

Név:	Homework8.zip
Feltöltés ideje:	2022. 11. 11. 0:27
Értékelés:	
Státusz:	Elfogadva
Feltöltések száma:	5
Értékelte:	Erdei Zsófia
Megjegyzések:	

Automatikus tesztelés eredményei

A megoldás átment a teszteken