

Funkcionális programozás

12. gyakorlat, 2019. 05. 07 – 08.

Algebrai adattípusok

1. Hozz létre egy `Day` adattípust, mely a hét napjait ábrázolja!
2. Írj egy `isFirstDayOfWeek` függvényt, mely eldönti, hogy egy nap a hét első napja-e!
3. Írj egy `isWeekend` függvényt, mely eldönti, hogy egy nap hétvégi nap-e!
4. Definiálj egy `Time` adattípust időpontokhoz (óra, perc)! Használj típuszsinonimákat ahol lehet!
5. Definiálj egy függvényt egy időpont szöveggé alakításához!

```
showTime (T 12 15) == "12.15"  
showTime (T 12 05) == "12.5"  
showTime (T 9 30)  == "9.30"
```

6. Definiálj egyenlőségvizsgálatot időpontok között!

```
eqTime (T 10 30) (T 10 30)  
not (eqTime (T 10 30) (T 10 40))  
not (eqTime (T 10 30) (T 11 30))  
not (eqTime (T 10 30) (T 9 30))
```

7. Vizsgáld meg, hogy egy időpont korábban van-e, mint egy másik!

```
isEarlier (T 10 15) (T 10 30)  
isEarlier (T 10 35) (T 12 30)  
not (isEarlier (T 12 30) (T 12 30))  
not (isEarlier (T 10 35) (T 10 30))  
not (isEarlier (T 13 30) (T 12 30))
```

8. Vizsgáld meg, hogy egy időpont két másik közé esik-e!
`isBetween t1 t2 t3`, ha `t2` időpont `t1` és `t3` közé esik (`t1 < t2 < t3` vagy `t3 < t2 < t1`).

```
isBetween (T 12 30) (T 12 40) (T 12 50)  
isBetween (T 12 50) (T 12 40) (T 12 30)  
not (isBetween (T 12 30) (T 12 15) (T 12 50))  
not (isBetween (T 12 30) (T 10 00) (T 12 50))  
not (isBetween (T 12 30) (T 15 00) (T 12 50))
```

9. Definiálj egy ún. smart constructort időpontokhoz! Ez hibát jelez érvénytelen időpontok esetén.

```
time 12 60 --> "time: invalid minute: 60"  
time 24 15 --> "time: invalid hour: 24"  
time (-5) 2 --> "time: invalid hour: -5"  
time 15 (-10) --> "time: invalid minute: -10"  
eqTime (time 12 30) (T 12 30)  
eqTime (time 23 59) (T 23 59)  
eqTime (time 00 00) (T 00 00)  
not (eqTime (time 12 30) (T 20 00))
```

10. Definiáld egy változatát az időpontnak, mely 12-órás am pm formában tárolja az időpontokat! Legyen ez `UStime`! Kérj hozzá egyenlőségvizsgálatot és szöveggé alakítást (`Eq` és `Show`)!

11. Definiáld, hogyan lehet egy `UStime` időpontot szöveggé alakítani!

```
showUStime (AM 10 15) == "10.15 am"  
showUStime (PM 2 30)  == "2.30 pm"
```

12. Alakíts át egy `UStime` időpontot `Time`-ra! 12.00 am éjfél (00.00), 12.00 pm delet jelent (a 12 óra a 12-órás időben 0 órának „felel meg”).

```
eqTime (usTimeToTime (AM 10 15)) (T 10 15)  
eqTime (usTimeToTime (AM 12 20)) (T 00 20)  
eqTime (usTimeToTime (PM 2 30)) (T 14 30)  
eqTime (usTimeToTime (PM 12 45)) (T 12 45)
```

13. Alakíts vissza egy `Time` időpontot `UStime`-ra!

```
timeToUStime (T 10 15) == AM 10 15  
timeToUStime (T 00 20) == AM 12 20  
timeToUStime (T 12 45) == PM 12 45  
timeToUStime (T 15 10) == PM 3 10
```