

Házi Feladat

A feladatokat egy **Homework1** nevű modulként kell beadni, a TMS a zip fájlokat fogadja el, így a kész kódot tömörítsétek zip-be, és úgy adjátok be. A feladatok végén zárójelben látható néven definiáljátok a függvényeket, kifejezéseket. Ügyeljete a helyes típuszignatúrára!

Első feladat - Függvényhasználat

Az alábbi kódrészletet másold be a fájlodba:

```
addThree :: Int -> Int
addThree x = x + 3

double :: Int -> Int
double x = x * 2

one :: Int -- "konstans függvény", olyan amelynek nincs bemeneti paramétere
one = 1

two :: Int
two = double one
```

A következő feladatokhoz csak a fenti 4 függvényt/kifejezést használd!

- Definiálj egy konstans kifejezést, amelynek `Int` a típusa és az értéke 4. (`four`)
- Definiálj egy függvényt, amely egy `Int`-et vár paraméterül és visszaad egy `Int`-et. A függvény duplázza meg a paraméterét majd adjon hozzá 3-at! (`doubleThenAddThree`)
- Definiálj egy konstans kifejezést, amelynek `Int` a típusa és az értéke 7. (`seven`)

Második Feladat - Mintaillesztés és If-Then-Else

- Definiálj egy függvényt, amely három `Int`-et kap paraméterül és egy `Int`-et ad vissza. Ha az első paraméter kisebb mint 0, adjuk vissza a második paraméterül kapott számot, egyébként pedig a harmadikat! (`cITE`)
- Definiálj egy függvényt, amely egy `Char`-t kap paraméterül és egy `Bool`-t ad vissza. A függvény adja vissza, hogy a paraméterül kapott karakter \$ szimbólum-e! (Használj mintaillesztést!) (`isDollar`)
- Definiálj egy függvényt, amely négy `Int`-et kap paraméterül és egy `Bool`-t ad vissza. Ha a paraméterül kapott számok szorzata pontosan 2180720, akkor adjunk vissza `True`-t egyébként `False`-ot! (`bigNumCheck`)
- Definiálj egy függvényt, amely két `Bool`-t kap paraméterül és egy `Bool`-t ad vissza. Végezzül el a paramétereken a logikai XOR (exkluzív vagy) műveletet. A függvényt implementáljuk háromszor: If-Then-Else-el, Mintaillesztéssel és már létező logikai függvényekkel! (`||`, `&&`, `not`) (`xor1`, `xor2`, `xor3`)