

## 10. házi feladat

### Kategória:

Házi feladatok

### Végső határidő:

5/11/2025, 11:59 PM (Beadva)

### Próbálkozások száma:

Korlátlan

### Kiírta:

Erdei Zsófia

### Leírás:

## Házi feladat

A házi feladatot egy Homework10 nevű modulként kell beadni. Minden definiálandó függvényhez adjuk meg a hozzá tartozó típuszignatúrát is! A feladatok után a zárójelben lévő név azt jelzi, milyen néven kell definiálni az adott függvényt, kifejezést. A forrásfájlban ügyeljetek arra, hogy minden kifejezés rendelkezzen helyes típuszignatúrával!

**Tipp:** Ha szükség van rá, a feladatokat bontsd fel részfeladatokra!

***A feladatok megoldásában nem lehet explicit rekurziót illetve listagenerátort használni!***

A teszteseteknél a következő példafájlon fognak futni a függvények:

```
testFile0 :: File
testFile0 = ["asd  qwe", "-- Foo", "", "\thello world "]
```

A fentebbi két sort is másoljátok be a beadott megoldásba.

A feladat megoldásához használjuk a Data.Char modult.

## Reprezentáció

Használjuk a következő típuszinonimákat.

```
type Line = String
type File = [Line]
```

## Számláló függvények

Adjuk meg azt a függvényt, amely megszámolja, hogy hány szó van egy sorban! Segítség: Használjuk a words függvényt!

```
countWordsInLine :: Line -> Int
```

Adjuk meg azt a függvényt, amely megszámolja, hogy hány szó van egy fájlban!

```
countWords :: File -> Int
```

Adjuk meg azt a függvényt, amely megszámolja, hogy hány karakter van egy fájlban!

```
countChars :: File -> Int
```

## Nagybetűsítés

Adjuk meg azt a függvényt, amely nagybetűsíti az összes szó első karakterét egy sorban! Segítség: Használjuk a words és unwords függvényeket!

```
capitalizeWordsInLine :: Line -> Line
```

## Transzformációk

Adjuk meg azt a függvényt, amely eldönti egy sorról, hogy az komment-e! Egy sor pontosan akkor számít kommentnek, ha a következő string-gel kezdődik: "--".

```
isComment :: Line -> Bool
```

Adjuk meg azt a függvényt, amely elhagyja egy fájlból a kommentezett sorokat!

```
dropComments :: File -> File
```

Adjuk meg azt a függvényt, amely egytől kezdődően megszámozza a sorokat! Tehát az n. sor a következő alakú lesz: `n: sor tartalma` .

```
numberLines :: File -> File
```

Adjuk meg azt a függvényt, amely elhagyja a sorvégekről a felesleges whitespace karaktereket! Segítség: Használjuk az `isSpace` függvényt!

```
dropTrailingWhitespaces :: Line -> Line
```

Adjuk meg azt a függvényt, amely egy tabulátort valahány szóközre cserél, minden egyéb karaktert pedig helyben hagy! Segítség: A tabulátort, mint karaktert a következőképpen jelöljük: `'\t'` .

```
replaceTab :: Int -> Char -> [Char]
```

Adjuk meg azt a függvényt, amely a fájlban a tabulátorokat valahány szóközre cseréli!

```
replaceTabs :: Int -> File -> File
```

## Tesztesetek

```
map countWordsInLine testFile0 == [2,2,0,2]
countWords testFile0 == 6
countChars testFile0 == 27
map capitalizeWordsInLine testFile0 == ["Asd Qwe","-- Foo","","Hello World"]
map isComment testFile0 == [False,True,False,False]
dropComments testFile0 == ["asd qwe","","\thello world "]
numberLines testFile0 == ["1: asd qwe","2: -- Foo","3: ","4: \thello world "]
dropTrailingWhitespaces "Hello world" == "Hello world"
dropTrailingWhitespaces "    Hello world" == "Hello world"
(map (replaceTab 3) $ concat testFile0) == ["a","s","d"," "," ","q","w","e","-","-"," ","F","o","o"," ","h","e","l","l","o"," ","w","o","r","l",
replaceTabs 5 testFile0 == ["asd qwe","-- Foo","","hello world "]
```

## Git tároló

### Útvonal:

<https://tms.inf.elte.hu/git/8737/v12198/w117012e2cd00946f0f55ccfac>

### Használat:

`git clone https://tms.inf.elte.hu/git/8737/v12198/w117012e2cd00946f0f55ccfac`

## Megoldás



### Név:

solution.zip

### Feltöltés ideje:

5/9/2025, 3:50 PM

### Értékelés:

### Státusz:

Elfogadva

### Feltöltések száma:

1

### Értékelte:

Szávó Tamás

### Megjegyzések:

#### Automatikus tesztelés eredményei

✔ #1