

Segédfüggvények, where

A továbbiakban ha használtok segédfüggvényt, az legyen az annak megfelelő `where/let...in...` blokkjában, mert nem kell, hogy mások lássanak a modulban esetleges "csúnya" kódokat (mert valamilyen előfeltétel mellett operálnak segédek).

1. Tömörítés

Hosszabb szövegeket kényelmesebb rövidebb módon tárolni. Ehhez készítsünk olyan függvényt `compress` néven, amely az egymás mellett szereplő azonos karaktereket rövidíti le az adott karakter és annak számára egy rendezett párként. A megoldásban ne használj még magasabb rendű függvényt!

Add meg a függvény lehető legáltalánosabb típusát!

Segítség: A `group :: Eq a => [a] -> [[a]]` függvény hasznos lehet ehhez, amely egymás mellett elhelyezkedő egyenlő elemeket csoportosít egybe. A `group` függvény a `Data.List` modulban található.

```
compress [] == ([] :: [(Char,Integer)])
compress "almafa" == [('a',1 :: Float),('l',1),('m',1),('a',1),('f',1),('a',1)]
compress "compress" == [('c',1 :: Integer),('o',1),('m',1),('p',1),('r',1),('e',1),('s',2)]
compress "sssszzzzziilllvaaaaaaa" == [('s',5),('z',6),('i',4),('l',2),('v',2),('a',8)]
compress [1 :: Integer,1,1,3,2,2,4,4,4,4] == [(1,3 :: Double),(3,1),(2,2),(4,4)]
compress [1 :: Double,1,1,3,2,2,4,4,3,4,4] == [(1,3 :: Int),(3,1),(2,2),(4,2),(3,1),(4,2)]
```

2. Szétválasztás választó elem mentén

Definiáld a `splitOn :: Eq a => a -> [a] -> [[a]]` függvényt, amely egy adott elem mentén bontja fel a paraméterül kapott listát.

```
splitOn 'A' "ÉnAelementemAaAvásárbaAfélApénzzel!" == ["Én","elementem","a","vásárba","fél","pénzzel!"]
splitOn 1 [1] == [[],[1]]
splitOn 1 [] == [[]]
splitOn "a" [] == [[]]
(take 50 $ splitOn 5 [1..] !! 1) == take 50 [6..]
```

3. Üres sorok

Készíts egy függvényt `emptyLines :: (Num a, Enum a) => String -> [a]`, ami megmondja egy szövegben, hogy hányadik sorok az üresek. Mivel a szövegekben (tehát nem a programozás világában) a sorszámozás 1-től kezdődik, így itt is kezdődjön az első sor az 1-es sorszámtól.

A megoldásban **ne** használj a `length` és `genericLength` függvényt!

```
emptyLines "elso\nmasodik\n\nnegyedik\n" == [3,5]
emptyLines "theme=dark\n\ntoolbar=0\n\nicons=gnome" == [2,4]
emptyLines "" == [1]
```

4. CSV fájl

Bontsd fel egy `csv` fájl tartalmát a celláira!

A sorok `'\n'`-ekkel, míg egy sorban az elemek `' '`-vel vannak egymástól elválasztva.

Nem kell konkrét fájl beolvasni, a fájlunkat egy `String` reprezentálja. Konkrét fájlbeolvasást csak haladó haskelles tudással lehet jól végrehajtani.

Add meg a függvény lehető legáltalánosabb típusát!

```
csv "nev,neptun,jegy\nEndre,ADG1K5,5\nAnna,KRJ25L,5" == [ ["nev","neptun","jegy"], ["Endre","ADG1K5","5"], [
```

Ha bármelyik függvény megírásához használtok segédfüggvényt, azokat tegyétek a megfelelő függvény `where`-hez vagy `let ... in ...`-hez tartozó részébe!

Tesztelő függvény

```
allPassed :: Bool
allPassed = null allTests

allTests :: [(String, Bool)]
allTests = [(str,test) |
  (str, test) <- [
    ("compress [] == ([ :: [(Char,Integer)])", compress [] == ([ :: [(Char,Integer)])])
  , ("compress \"almafa\" == [('a',1 :: Float),('l',1),('m',1),('a',1),('f',1),('a',1)]", compress "a
  , ("compress \"compress\" == [('c',1 :: Integer),('o',1),('m',1),('p',1),('r',1),('e',1),('s',2)]",
  , ("compress \"sssssszzzzziillvvaaaaaaa\" == [('s',5),('z',6),('i',4),('l',2),('v',2),('a',8)]",
  , ("compress [1 :: Integer,1,1,3,2,2,4,4,4,4] == [(1,3 :: Double),(3,1),(2,2),(4,4)]", compress [1
  , ("compress [1 :: Double,1,1,3,2,2,4,4,3,4,4] == [(1,3 :: Int),(3,1),(2,2),(4,2),(3,1),(4,2)]", co
  , ("splitOn 'A' \"ÉnAelmentemAaAvásárbaAfélApénzzel!\" == [\"Én\", \"elmentem\", \"a\", \"vásárba\", \"
  , ("splitOn ',' \"12,23,34,\" == [\"12\", \"23\", \"34\", \"\"], splitOn ',' \"12,23,34,\" == [\"12\", \"23
  , ("splitOn (2 :: Integer) [1,2,2,3,2] == [[1],[],[3],[ ]]", splitOn (2 :: Integer) [1,2,2,3,2] == [
  , ("splitOn 1 [1] == [[],[ ]]", splitOn 1 [1] == [[],[ ]])
  , ("splitOn 1 [] == [[]]", splitOn 1 [] == [[]])
  , ("splitOn \"a\" [] == [[]]", splitOn "a" [] == [[]])
  , ("splitOn False [False, True, True, False, False, True] == [[],[True, True],[],[True]]", splitOn
  , ("take 10 (splitOn (1 :: Int) (repeat 1)) == [[],[ ],[ ],[ ],[ ],[ ],[ ],[ ],[ ],[ ],[ ]]", take 10 (splitOn (
  , ("take 8 (splitOn 2 (cycle [1,2,3,4])) == [[1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3
  , ("take 50 (splitOn 5 [1..] !! 1) == take 50 [6..]", take 50 (splitOn 5 [1..] !! 1) == take 50 [6.
  , ("emptyLines \"elso\nmasodik\nnegyedik\n\" == [3 :: Int,5]", emptyLines "elso\nmasodik\nnegye
  , ("emptyLines \"theme=dark\nntoolbar=0\nnicons=gnome\" == [2 :: Integer,4]", emptyLines "theme=d
  , ("emptyLines \"\" == [1 :: Double]", emptyLines "" == [1 :: Double])
  , ("emptyLines \"\n\nKicsit lentebb kezdődik\na sor\n\n\" == [1,2,5,6]", emptyLines "\n\nKicsit len
  , ("take 50 (emptyLines (repeat '\n')) == take 50 [1..]", take 50 (emptyLines (repeat '\n')) == tak
  , ("null (emptyLines \"alma\")", null (emptyLines "alma"))
  , ("null (emptyLines \"alma\nkörte\nszilva\")", null (emptyLines "alma\nkörte\nszilva"))
  , ("csv \"nev,neptun,jegy\nEndre,ADG1K5,5\nAnna,KRJ25L,5\" == [\"nev\", \"neptun\", \"jegy\"],[\"End
  , ("csv \"\n\nmegtett km,fogyasztas,\n745,7.1 l/100 km\n800,,megj:nem sikerult lemerni\n796,6.5 l/1
  , ("null (csv [])", null (csv []))
  ], not test]
```

Összes teszt

```
compress [] == ([] :: [(Char,Integer)])
compress "almafa" == [('a',1 :: Float),('l',1),('m',1),('a',1),('f',1),('a',1)]
compress "compress" == [('c',1 :: Integer),('o',1),('m',1),('p',1),('r',1),('e',1),('s',2)]
compress "sssszzzzzziiiiillvvaaaaaaa" == [('s',5),('z',6),('i',4),('l',2),('v',2),('a',8)]
compress [1 :: Integer,1,1,3,2,2,4,4,4,4] == [(1,3 :: Double),(3,1),(2,2),(4,4)]
compress [1 :: Double,1,1,3,2,2,4,4,3,4,4] == [(1,3 :: Int),(3,1),(2,2),(4,2),(3,1),(4,2)]
splitOn 'A' "ÉnAelmentemAaAvásárbaAfélApénzzel!" == ["Én","elmentem","a","vásárba","fél","pénzzel!"]
splitOn ', ' "12,23,34," == ["12","23","34",""]
splitOn (2 :: Integer) [1,2,2,3,2] == [[1],[],[3],[ ]]
splitOn 1 [1] == [[],[ ]]
splitOn 1 [] == [[]]
splitOn "a" [] == [[]]
splitOn False [False, True, True, False, False, True] == [[],[True, True],[],[True]]
take 10 (splitOn (1 :: Int) (repeat 1)) == [[],[],[],[],[],[],[],[],[ ],[ ]]
take 8 (splitOn 2 (cycle [1,2,3,4])) == [[1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3,4,1],[3,4,1]]
take 50 (splitOn 5 [1..] !! 1) == take 50 [6..]
emptyLines "elso\nmasodik\n\nnegyedik\n" == [3 :: Int,5]
emptyLines "theme=dark\n\ntoolbar=0\n\nicons=gnome" == [2 :: Integer,4]
emptyLines "" == [1 :: Double]
emptyLines "\n\nKicsit lentebb kezdődik\na sor\n\n" == [1,2,5,6]
take 50 (emptyLines (repeat '\n')) == take 50 [1..]
null (emptyLines "alma")
null (emptyLines "alma\nkörte\nszilva")
csv "nev,neptun,jegy\nEndre,ADG1K5,5\nAnna,KRJ25L,5" == [["nev","neptun","jegy"],["Endre","ADG1K5","5"],[
csv "\n\nmegtett km,fogyasztas,\n745,7.1 l/100 km\n800,,megj:nem sikerult lemerni\n796,6.5 l/100 km,,\n\n
null (csv [])
```