Funkcionális programozás elméleti kvíz  
Nem hivatalos teszt sor 1 (Heki Z12 modellje alapján )  
**FIGYELEM:** Ez a feladatsor gyakorló célokat szolgál, és 7 kérdésből áll. A megoldás során minden kérdésnél akár több helyes válasz is létezhet. A jelölés legyen egyértelmű (karikázás vagy X).

**1. Melyik kifejezéssel ekvivalens az alábbi listakifejezés?***[f x | x <- xs, g x]*

1. (map f . filter g) xs
2. filter g . map f) xs
3. (filter f . map g) xs
4. (map g . filter f) xs

**2. Melyik kifejezés típusozható az alábbiak közül?**

1. "hello" ++ 5
2. length [1, 2, 3] + "test"
3. take 2 [1, 2, 3, 4]
4. A felsoroltak egyike sem.

**3. Mi a foldl függvény típusa?**

1. (b -> a -> b) -> b -> [a] -> b
2. (a -> b -> b) -> b -> [a] -> b
3. (b -> a -> b) -> [a] -> b -> b
4. (a -> b -> b) -> [a] -> b -> b

**4. Az alábbi definíciók közül melyik ad meg totális függvényt?**

1. f (x:xs) = x
2. f [] = undefined
3. f (Just x) = x
4. f x = x

**5. Mi nem lehet a típusa a Monday adatkonstruktornak az alábbiak közül?***data SomeDays a b = Monday a b | Sunday*

1. Int -> Bool -> SomeDays Int Bool
2. Char -> String -> SomeDays Char String
3. Int -> Int -> SomeDays Bool Int
4. Bool -> String -> SomeDays Bool String

**6. Melyik kifejezés illeszkedik az** *(x:y):z* **mintára?**

1. [ [1, 2], [3, 4] ]
2. (1, [2, 3])
3. [ "ab", "cd" ]
4. [[1, 2], 3]

**7. Az alábbi függvények közül melyiknek lehet totális definíciót adni?**

1. f :: [Int] -> Int
2. g :: [a] -> Maybe a
3. h :: (a -> a) -> [a]
4. i :: a -> a

**Helyes válaszok:**

1. A
2. C
3. A
4. D
5. C
6. A és C
7. B és D