

Válasszuk ki a helyes választ a megadottak közül! Minden kérdésnél egyetlen helyes válasz létezik, javított megoldás nem fogadható el. 10 helyes válasz szükséges az elméleti teszt teljesítéséhez.

1. Az alábbi állítások közül melyik igaz a Haskell nyelvre?

- a) Haskellben kötelező megadni minden kifejezés típusát.
- b) Haskellben nem lehet operátorokat definiálni, csak a beépítetteket lehet használni.
- c) A Haskell egy mohó kiértékelésű, tisztán funkcionális nyelv.
- d) A Haskell egy lusta kiértékelésű, tisztán funkcionális nyelv.

2. Melyik függvény definíciója totális (nem parciális) az alábbiak közül?

- a) tail
- b) init
- c) last
- d) take

3. Melyik állítás hamis az alábbiak közül?

- a) Ha egy függvény rekurzív, közvetve vagy közvetlenül magára hivatkozik.
- b) A foldr rekurzívan definiált függvény.
- c) Egy rekurzív függvénynek tetszőleges számú paramétere lehet.
- d) Minden rekurzív függvénynek meg kell adni legalább egy alapesetet.

4. Mi igaz a lambda függvényekre Haskellben?

- a) Új típus bevezetésével definiálhatók.
- b) Magasabbrendű függvényeknek adhatók paraméterként.
- c) A kompozíciós műveletek egy másik elnevezése.
- d) A lambda kifejezéseknek nem vezethető le vagy adható meg a típusa.

5. Melyik típusosztály a következők közül?

- a) Int
- b) Bool
- c) Integral
- d) Double

6. Melyik kifejezés helyes nyelvtanilag az alábbiak közül?

- a) 8 mod 5
- b) 3 `mod` 5
- c) 6 (mod) 6
- d) (1) mod (7)

7. Mi az 1 < (1,2) kifejezés eredménye?

- a) Fordítási idejű hiba.
- b) True
- c) False
- d) Futási idejű hiba.

8. Mít nevezünk részleges függvényalkalmazásnak?

- a) Amikor egy parciális függvényt egy olyan paraméterre hívunk meg, amire nem lett definiálva.
- b) Amikor egy függvényt átadunk paraméterül egy másiknak.
- c) Amikor egy függvény önmagára hivatkozik, hogy egy részfeladatot megoldjon.
- d) Amikor egy `n` darab paraméterrel rendelkező függvénynek kevesebb, mint `n` paramétert adunk át.

9. Mikor mondjuk egy programozási nyelvről, hogy tisztán funkcionális?

- a) Ha magasabbrendű függvények használhatók benne.
- b) Ha nincsenek mellékhatások.
- c) Ha megtalálható benne a Curry-módszer.
- d) Ha lusta kiértékelésű.

10. Melyik lehet a map (even . (+1)) kifejezés típusa?

- a) Integral a => (a -> a) -> [a] -> [a]
- b) Integral a => [a] -> [Bool]
- c) Integral a => [a] -> [a]
- d) [a] -> [Bool]

11. Az alábbi kifejezések közül melyik ekvivalens az [f x | x <- 1s] kifejezéssel véges, hiba nélkül kiértékelhető 1s lista esetén?

- a) foldr f [] 1s
- b) map f 1s
- c) filter f 1s
- d) any f 1s

12. Mi a (\x -> x:[x]) kifejezés típusa?

- a) a -> [[a]]
- b) a -> a
- c) a -> [a]
- d) [a] -> a

13. Mi az eredménye a length []:[] kifejezésnek?

- a) 0
- b) [0]
- c) Fordítási idejű hiba
- d) Futási idejű hiba

14. Mi a különbség az \_ és x minták közt?

- a) Az `x` csak a lista fejelemeként, az `\_` pedig bárhol használható a mintában.
- b) Az `\_` akkor használandó, ha a típus deklarációban adott típusoktól eltérő értékeket is kezelni szeretnénk.
- c) Egy mintában az `\_` többször is szerepelhet, míg az `x` legfeljebb egyszer.
- d) Egy mintában az `\_` többször is szerepelhet, a függvény törzsében véletlenszerű értéket képvisel.

15. Mit jelent az (xs:x:y) minta?

- a) Tetszőleges listára illeszkedik.
- b) Egy legalább kételemű listára illeszkedik.
- c) Pontosan három elemből álló listára illeszkedik.
- d) Legalább kételemű lista, melynek utolsó két eleme lesz az `x` és `y`.

16. Melyik kifejezésre illeszkedik a []:x minta?

- a) [[1], [2]]
- b) [[]]
- c) []
- d) [[1,2]]

17. Mi az eredménye az alábbi kifejezésnek?

filter (== 2) ([1..] ++ [1,2])

- a) [2,2]
- b) Kiír egy kettest, majd végtelen ideig fut.
- c) Fordítási idejű hiba.
- d) Nem áll le, nem ír ki semmit.

18. Melyik állítás igaz Haskellben a listákra?

- a) A listák heterogének, azaz különböző típusú elemekből is állhatnak.
- b) A listák homogének, azaz kizárólag azonos típusú elemekből állhatnak.
- c) A listák mindenképpen véges hosszúak.
- d) Minden listát csak egyféleképpen lehet megkonstruálni, a `( : )` operátorral.

19. Az alábbiak közül melyik nem lehet a Just "true" kifejezés típusa?

- a) Maybe [Char]
- a) Maybe String
- c) Maybe Bool
- d) A fentiek közül bármelyik lehet.

20. Tekintsük az alábbi definíciót! Mi a B 42 kifejezés típusa?

data D = A Int | B Int [Char]

- a) D
- b) String -> D
- c) B Int
- d) D -> String

Válaszok

- 1-10: D D D B C B A D B B
- 11-20: B C B C B B B C B