

Контролно упражнение № 1 по Функционално програмиране (примерен вариант)

Име фак. № адм. група

1. Дайте примери за поне три типа образци (patterns) в езика Haskell. Обяснете кога съответните аргументи са съпоставими с тези образци.
2. Дайте пример за дефиниция на функция, в която се използва обща рекурсия върху списъци.
3. Кои от следните конструкции са коректно дефинирани списъци в Haskell?
 - a) ["A", 'A']
 - b) 'A' : []
 - c) [(123, "Hello"), (123, "Hello", "World")]
 - d) [{"123", "Hello"}, {"123", "Hello", "World"}]
4. Кои от следните конструкции са коректно дефинирани (валидни) списъци в Haskell? Запишете валидните списъци с помощта на ":" нотацията.
[1, 2, 3, []]
[1, [2, 3], 4]
[[1, 2, 3], []]
5. Какъв е типът на конструкцията ("Worl", 'd', "123", 456)?
6. Нека е даден списък **lst** от тип **[(Char), Int, Float]**, съдържащ данни за имената, факултетните номера и успеха на студентите от една група. Напишете изрази на Haskell, с помощта на които се получават:
 - a) списък от имената на студентите от групата
 - b) броят на отличните студенти в групата
7. Попълнете липсващите изрази в дефиницията на функцията **merge** при условие, че тя слива два списъка от цели числа **l1** и **l2**, които са сортирани във възходящ ред (резultatът също е сортиран във възходящ ред и съвпадащите елементи на **l1** и **l2** участват в него в толкова екземпляра, колкото пъти се срещат сумарно в **l1** и **l2**):
merge :: [Int] -> [Int] -> [Int]
merge [] l2 = l2
merge l1 [] = l1
merge (l1:ls1) (l2:ls2)
| l1 <= l2 =
| otherwise =
8. Дефинирайте функция на Haskell, която премахва всички кратни на 3 числа от даден списък от тип **[Int]**.

9. Дефинирайте функция на Haskell, която проверява дали поредните елементи на списъка от числа **lst** образуват монотонно растяща редица. Пример за такъв списък е [1,5,6,6,10].
10. Дефинирайте функция на Haskell, която намира сумата на два полинома, зададени във вида $[(<коэф.{}_1>, <степен{}_1>), (<коэф.{}_2>, <степен{}_2>), \dots, (<коэф.{}_n>, <степен{}_n>)]$, където всички коефициенти са различни от 0 реални числа и степенните показатели са подредени в строго намаляващ ред (резултатът трябва да има същия вид).

Задача	Точки
1	0,3
2	0,3
3	0,3
4	0,4
5	0,2
6	0,6
7	0,8
8	0,6
9	0,6
10	1,4