**Контролно № 3 по ФП**

**Специалност „Компютърни науки“, 22.01.2015 г.**

**Вариант 1**

**Задача 1.** Да се напише функция *isPalindrome n* на езика Haskell, която проверява дали цифрите на цялото неотрицателно число *n* образуват палиндром.

*Пример*:

isPalindrome 12321 - True

isPalindrome 4544 - False

isPalindrome 0 - True

isPalindrome 7870 - False

**Задача 2.** Да се напише функция*frequency dictionary* на езика Haskell, която за даден списък от цели числа генерира честотен речник.

*Пример*: За списъка [12, 8, 17, 12, 4, 8, 3, 17, 8, 4, 12, 8], да се генерира честотният речник [[12, 3], [8, 4], [17, 2], [4, 2], [3, 1]].

*Бонус точки* – ако в резултатния списък числата да са подредени по честотност – от най-често срещаното число към най-рядко срещаното.

**Задача 3.** Да се напише функция *variations k l p q* на езика Haskell, която намира всички вариации без повторения на *k* елемента на списъка от цели числа *l*, разликата от елементите на които принадлежи на интервала [*p*, *q*].

*Пример*: variations2 [10, 20, 30] -15 15 = [ [10, 20], [20, 10], [20, 30], [30, 20]]. Всички вариации без повторения на елементите на списъка [10, 20, 30] при k = 2 са [ [10, 20], [10, 30], [20, 10], [20, 30], [30, 10], [30, 20]].

*Забележкa*: *k* е не по-голямо от дължината на списъка *l*.

**Контролно № 3 по ФП**

**Специалност „Компютърни науки“, 22.01.2015 г.**

**Вариант 2**

**Задача 1.** Симетрично на дадено положително цяло число се нарича число със същите цифри, но записани в обратен ред*.* Да се напише функция *symNumber n* на езика Haskell, която намира симетричното число на дадено естествено число.

*Пример*:

symNumber1232 връща 2321

symNumber4567 връща 7654

symNumber 6 връща 6

**Задача 2.** Дадени са два списъка от цели числа *a* и *b*. Да се дефинира функция*frequency dictionary* на езика Haskell, която за елементите на списъка *a* намира броя на срещанията им в списъка *b*.

*Пример*: Ако a = [2, 3, 4, 2, 6] и b = [1, 4, 2, 3, 4, 5, 4, 2, 2, 4, 4, 7], да се генерира списъкът [[2, 3], [3, 1], [4, 5], [6, 0]].

*Бонус точки* – ако в резултатния списък числата да са подредени по честотност – от най-често срещаното число към най-рядко срещаното.

**Задача 3.** Да се напише функция *variations k l p q* на езика Haskell, която намира всички вариации без повторения на *k* елемента на списъка от цели числа *l*, сумата от елементите на които принадлежи на интервала [*p*, *q*].

*Пример*: variations2 [10, 20, 30] 35 45 = [[10, 30], [30, 10]]. Всички вариации без повторения на елементите на списъка [10, 20, 30] при k = 2 са [ [10, 20], [10, 30], [20, 10], [20, 30], [30, 10], [30, 20]].

*Забележкa*: *k* е не по-голямо от дължината на списъка *l*.