

С помощта на map, filter и foldr да се напишат функции:

1. countNames, която по даден списък от английски думи намира броя на собствените имена в тях (тези, които започват с главна буква)
2. strangeSum l, която по даден списък l от цели числа намира сумата от квадратите на четните числа и кубовете на нечетните

3. $\sum_{i=1}^n x^i y^{n-i}$
calcSum1 n x y, която пресмята

4. minProdPosList ll, която по списък от списъци от числа ll, намира произведението на минималните елементи на тези от списъците в ll, които се състоят само от положителни числа

5. $\sum_{i=1}^n \prod_{j=i}^m (x^i - y^j)$
calcSum2 n m x y, която пресмята

6. * calcPoly l x, която по полином, представен чрез списък от коефициентите си, пресмята неговата стойност в точката x по метода на Хорнер: ((a0*x+a1)*x+a2)*x + ... an (упътване: използвайте foldr)

Оценка на студент по даден предмет се задава с наредена тройка (факултетен номер, име на предмет, оценка). Дадени са типовете

```
type Grade = (Int, String, Double)
type DB = [Grade]
```

Всеки студент може да има най-много една оценка по даден предмет.

Да се напишат функции

1. averageGrade :: DB -> Double, която намира средната от всички оценки
2. listStudents :: DB -> [Int], която намира списък от всички студенти с оценки
3. averageStudentGrade :: DB -> Int -> Double, която намира средната оценка на даден студент
4. listFailed :: DB -> String -> [Int], която намира списък с всички студенти, скъсани по даден предмет
5. bestStudent :: DB -> Int, която намира студента с най-добър среден успех от всички предмети
6. worstSubject :: DB -> String, която намира предмета, по който средния успех е най-нисък