

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»

Институт компьютерных наук и технологий

Высшая школа искусственного интеллекта

Дисциплина:
ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ

ПРАКТИЧЕСКОЕ ЗАДАНИЕ 1

Обучающийся гр. 3530201/10001

Нгуен Куок Дат

Руководитель

Моторин Дмитрий Евгеньевич

Санкт-Петербург 2022

Содержание

1	Реализация	2
1.1	Задание 1	2
1.2	Задание 2	3
1.3	Задание 3	3
1.4	Задание 4	4
1.5	Задание 5	5

1 Реализация

1.1 Задание 1

Реализовать функцию

$$\prod_{l=1}^n \sum_{k=1}^l \prod_{p=1}^k \sum_{i=1}^p i$$

со значением $n = 3$.

При запуске функции используя команду “:set +s”, записывать в отчет время выполнения и количество затрачиваемой памяти.

Надо реализовать 4 рекурсивных функций: сложение до p **addp: Int->Int**, умножение до k **mulk: Int->Int**, сложение до l **addl: Int->Int**, умножение до n **muln: Int->Int**. Результат задания - вывод функции **muln**

```
addp :: Int->Int
addp 1 = 1
addp p = p + addp (p - 1)

mulk :: Int->Int
mulk 1 = 1
mulk k = addp k * mulk (k - 1)

addl :: Int->Int
addl 1 = 1
addl l = mulk l + addl (l - 1)

muln :: Int->Int
```

```

muln 1 = 1
muln n = add1 n * muln (n - 1)

```

1.2 Задание 2

Реализовать функцию, разворачивающую список $S1 = [kiveydewhjusgofimbbyhwbopvuplwfexresmhntnic]$ ($[1,2,3] \rightarrow [3,2,1]$).

Реализуют функцию **reverse1: String->String**, в которой рекурсивно перемещают голову строки в конец.

```

reverse1 :: String -> String
reverse1 [] = []
reverse1 a = (++) (reverse1 (tail a)) [head a]

```

1.3 Задание 3

Реализовать функцию конкатенации списка списков $S2 = [[kiv], [eydew], [hjusg], [mbbyhwbopvupl], [ofi], [wfexresmhntnic]]$ ($[[1], [2], [3], [4]] \rightarrow [1,2,3,4]$)

Реализуют функцию **concat1: [String]->String**, в которой рекурсивно добавляют каждую данную строку в одну другую.

```

concat1 :: [String] -> String
concat1 [] = []
concat1 a = (++) (head a) (concat1 (tail a))

```

1.4 Задание 4

Реализовать функцию, которая вставляет строку “-,-” между списками S_1 и S_2 .

Реализуют функцию `separate: [String] -> String`, в которой рекурсивно конкатенируют в порядке элементарную строку, “-,-” в начало результирующей строки.

```
separate:: [String] -> String
separate [] = []
separate a = concat1 [head a, "-,-", separate (tail a)]
```

1.5 Задание 5

Реализовать вычисление элемента треугольника Паскаля (расчет i -ого числа в j -ой строке).

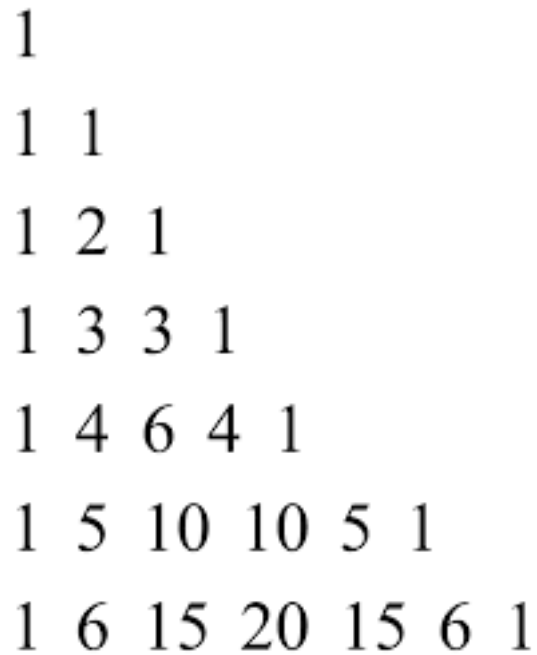


Рис. 1: Треугольник Паскаля

Формула вычисления число Паскаля на строку j и столбец i :

$$p(0,0) = p(0,i) = p(i,i) = 1$$

$$p(i,j) = p(i-1,j-1) + p(i,j-1)$$

Реализуют функцию **pascal: (Int,Int)->Int**, в которой рекурсивно вычисляют число Паскаля с данным положением.

```
pascal :: Int->Int->Int
pascal 0 0 = 1
pascal 0 a = 1
pascal i j =
    if i == j then 1
    else pascal (i-1) (j-1) + pascal i (j-1)
```

1.6 Задание 5

1.6.1 Математическое описание

Фрактал *Реализовать вычисление элемента треугольника Паскаля (расчет i -ого числа в j -ой строке).*

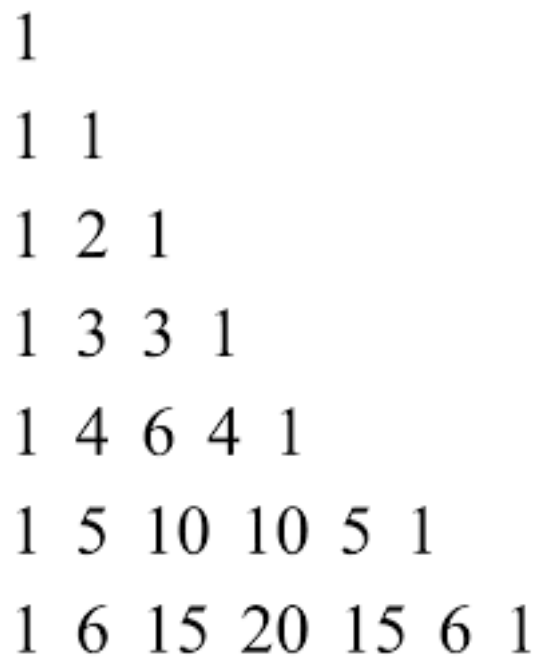


Рис. 2: Треугольник Паскаля

Формула вычисления число Паскаля на строку **j** и столбец **i**:

$$p(0,0) = p(0,i) = p(i,i) = 1$$

$$p(i,j) = p(i-1,j-1) + p(i,j-1)$$

Реализуют функцию **pascal: (Int,Int)->Int**, в которой рекурсивно вычисляют число Паскаля с данным положением.


```
pascal :: Int->Int->Int
pascal 0 0 = 1
pascal 0 a = 1
pascal i j =
    if i == j then 1
    else pascal (i-1) (j-1) + pascal i (j-1)
```