ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ

(МТУСИ)



Лабораторная работа №4

Введение в Информационные технологии

Выполнил студент

группы БВТ1903:

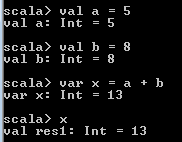
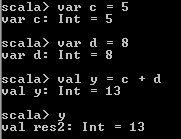
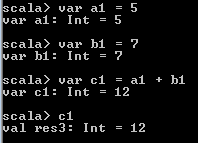
Игнатов Д.В.

­

*Москва,2021*

Ответы на вопросы

1. Переменные res – это значения val или настоящие переменные var?

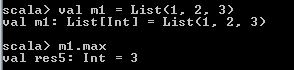
Из примеров видно, что несмотря на значения, используемые в операциях, res – это значения val.

1. "crazy" \* 3 в REPL

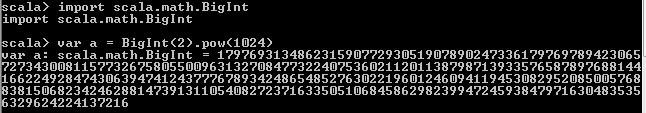


1. Что означает выражение 10 max 2? В каком классе определен метод max?

Метод max () используется для поиска наибольшего элемента из всех элементов в указанном списке. Класс Math?

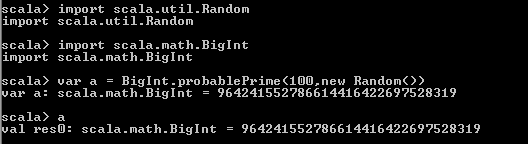


1. Используя число типа BigInt, вычислите 2^1024



1. Что нужно импортировать, чтобы найти случайное простое число вызовом метода probablePrime(100, Random) без использования каких-либо префиксов перед именами probablePrime и Random?

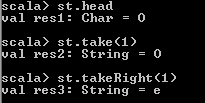
Import scala.math.BigInt



1. Один из способов создать файл или каталог со случайным именем состоит в том, чтобы сгенерировать случайное число типа BigInt и преобразовать его в систему счисления по основанию 36, в результате получится строка, такая как "qsnvbevtomcj38o06kul". Отыщите в Scaladoc методы, которые можно было бы использовать для этого.



1. Как получить первый символ строки в языке Scala? А последний символ?



1. Что делают строковые функции take, drop, takeRight и dropRight? Какие преимущества и недостатки они имеют в сравнении с substring?

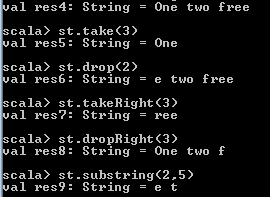
**take(n:Int)** Строка, содержащая первые n символов этой строки. Получить строку от первого до n-го символа.

**drop (n:Int)** Остальная часть строки без первых n символов. Возвращает строку без первых n символов.

**takeRight(n: Int)** Строка, содержащая последние n символов этой строки.

**dropRight(n: Int)** Остальная часть строки без последних n символов. Вернуть строку без последних n символов.

**substring(start: Int, end: Int) извлекает символы от start до end-1**



1. Сигнум числа равен 1, если число положительное. -1 – если отрицательное, и 0 – если равно нулю. Напишите функцию, вычисляющую это значение.

*object Task{*

*def main(args: Array[String]){*

*println("Input x");*

*var x = scala.io.StdIn.readInt();*

*if(x<0) {*

*println("SGN(x) = -1")*

*}*

*if(x==0) {*

*println("SGN(x) = 0")*

*}*

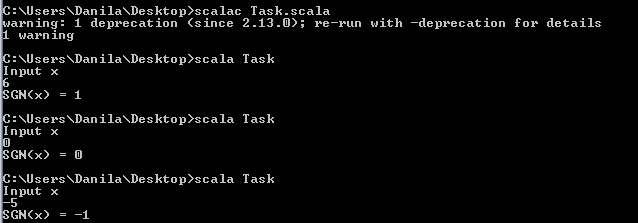
*if(x>0) {*

*println("SGN(x) = 1")*

*}*

*}*

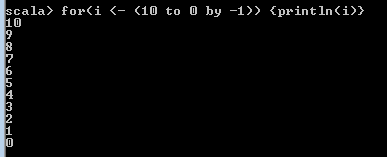
*}*



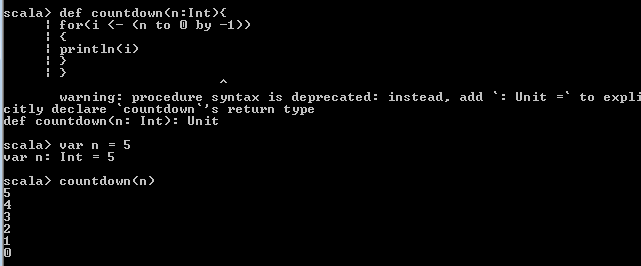
1. Какое значение возвращает блок {}? Каков его тип?



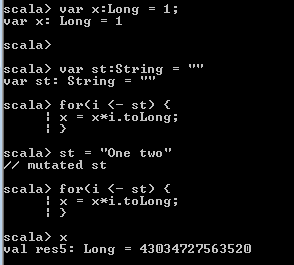
1. Напишите на языке Scala цикл, эквивалентный циклу на языке Java for (int i=10; i>=0; i--) System.out.println(i);



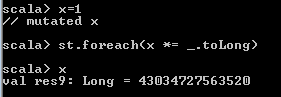
1. Напишите процедуру countdown (n: Int), которая выводит числа от n до 0



1. Напишите цикл for для вычисления кодовых пунктов Юникода всех букв в строке. Например, произведение символов в строке «Hello» равно 9415087488L.

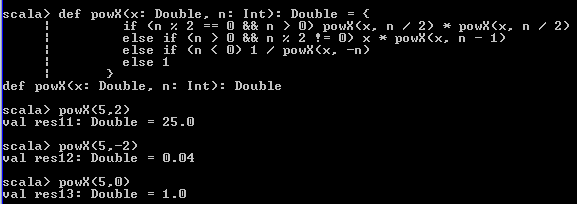


1. Решите предыдущее упражнение без применения цикла. Напишите функцию product(s: String), вычисляющую произведение, как описано в предыдущих упражнениях.

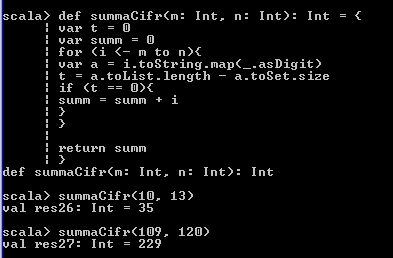


16. Сделайте функцию из предыдущего упражнения рекурсивной.

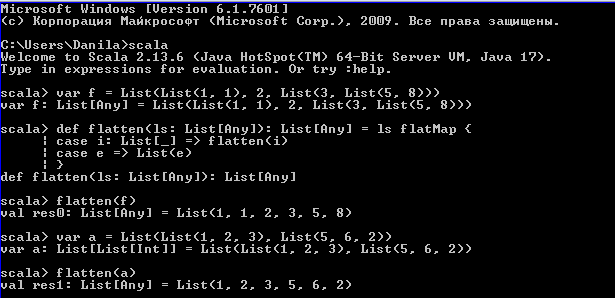
17.Напишите функцию, вычисляющую xn, где n – целое число. Используйте следующее рекурсивное определение: • xn = y2, если n – четное и положительное число, где y=xn/2 • xn = x\*xn-1, если n – нечетное и положительное число. • x0=1. • xn=1/x-n, если n – отрицательное число. Не используйте инструкцию return.



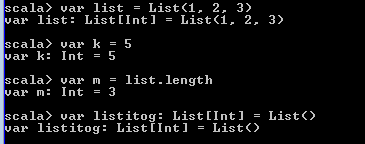
18. f(m,n) - сумма всех натуральных чисел от m до n включительно, в десятичной записи которых нет одинаковых цифр.

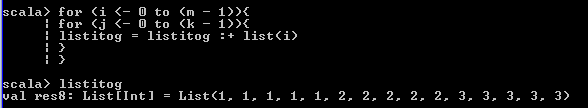


19. Список содержит целые числа, а также другие списки, такие же как и первоначальный. Получить список, содержащий только целые числа из всех вложенных списков. Пример: f(List(List(1, 1), 2, List(3, List(5, 8)))) = List(1, 1, 2, 3, 5, 8)

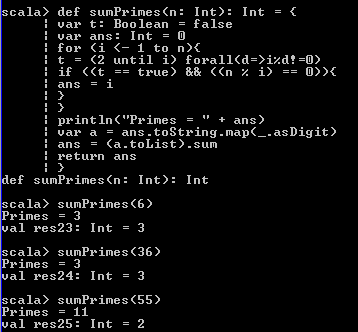


21. 23. Список содержит элементы одного, но любого типа. Получить список, содержащий каждый имеющийся элемент старого списка k раз подряд. Число k задается при выполнении программы.

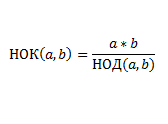




20, 22. f(n) - сумма цифр наибольшего простого делителя натурального числа n.



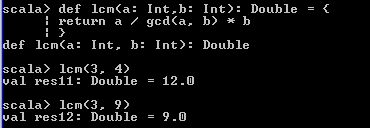
1. f(m,n) - наименьшее общее кратное натуральных чисел m и n.



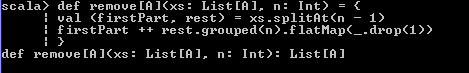
НОД

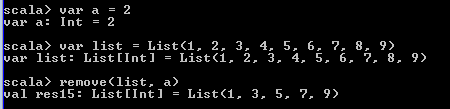


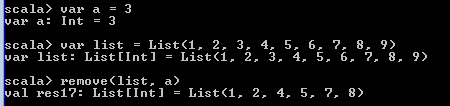
НОК



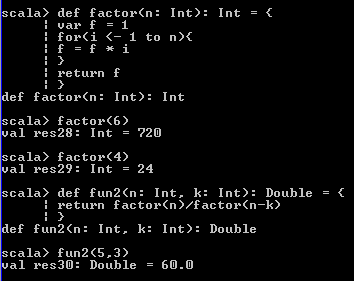
1. Список содержит элементы одного, но любого типа. Получить список, из элементов исходного, удаляя каждый k-й элемент. Число k задается при выполнении программы.







26. f(n,k) - число размещений из n по k. Факториал не использовать.



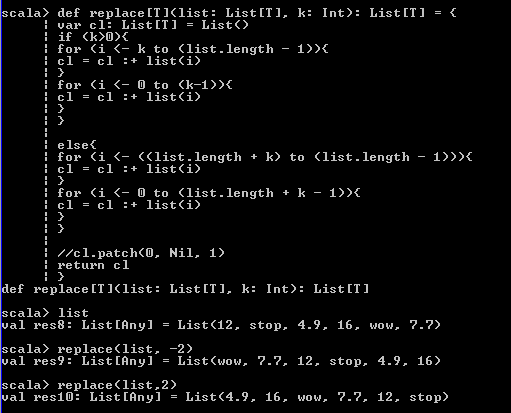
27. Список содержит элементы одного, но любого типа. Получить

новый список, перемещая циклически каждый элемент на k позиций влево

(при перемещении на одну позицию первый элемент становится последним,

второй первым и так далее). Число k задается при выполнении программы.

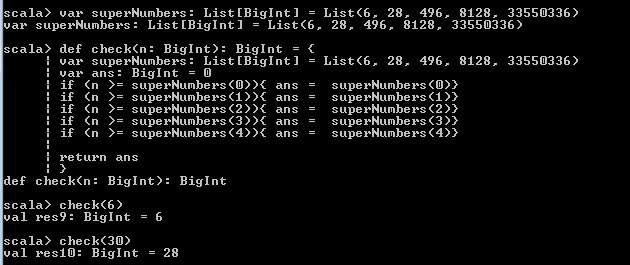
Если k отрицательное, то перемещение происходит вправо.



28. f(n) - наибольшее совершенное число не превосходящее n.

Совершенным называется натуральное число n равное сумме своих

делителей, меньших n, например 6 = 1 + 2 + 3 ( f(6) = 6, f(7) = 6, ... ).

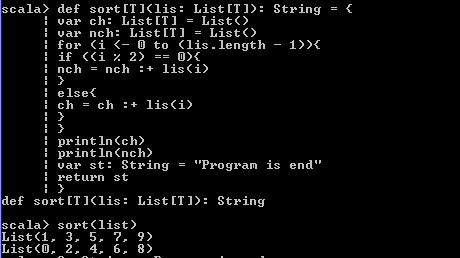


29. Список содержит элементы одного, но любого типа. Получить

два списка из элементов исходного, выбирая в первый элементы с четными

индексами, а во второй с нечетными.

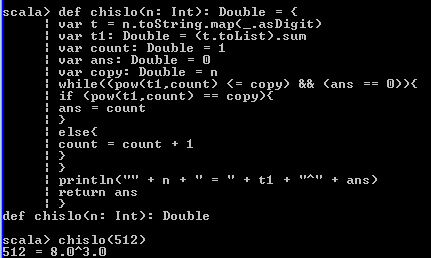




30. f(n) - наибольшее из чисел от 1 до n включительно, обладающее

свойством: сумма цифр n в некоторой степени > 1 равна самому числу n.

Пример: 512 = 83



31. Список в качестве элементов содержит кортежи типа: (n, s), где n

— целые числа, а s — строки. Получить два списка из элементов исходного,

выбирая в первый числа, а во второй строки из кортежей.

