|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **БИОМЕДИЦИНСКАЯ ТЕХНИКА**

КАФЕДРА **БИОМЕДИЦИНСКИЕ ТЕХНИЧЕСКИЕ СИСТЕМЫ (БМТ-1)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.03 Прикладная информатика (Цифровые биомедицинские системы)**

**Отчет**

|  |
| --- |
| **по лабораторной работе № 2** |

**Название:**

Коллекции и строки

**Дисциплина:** Алгоритмизация и программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | БМТ1-13Б |  |  | В.А. Герасимов |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Т.А.Ким |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

# Задание 1

Определить число инверсий в массиве из n ≤ 20 целых чисел (инверсия – это пара элементов, в которой большее число расположено слева от меньшего).

# Исходный код

using Test

function IsInverse(fir\_n, sec\_n)

  if fir\_n > sec\_n

    return true

  else

    return false

  end

end

function InverseCount(array)

  inv\_count = 0

  for i in range(2, length(array)) # Основной цикл посчета количества инверсий

    if IsInverse(array[i-1], array[i])

      inv\_count += 1

    end

  end

  return inv\_count

end

println("Введите подряд через пробел 20 чисел: ") # Задаем массив чисел

n = split(readline())

n = map(x->parse(Int64, x), n)

println(InverseCount(n))

@testset "IsInverseTest" begin

  @test IsInverse(9, 5) == true

  @test IsInverse(5,9) == false

end

@testset "InverseCountTest" begin

  @test InverseCount([1,2,3,4,5,7,6]) == 1

  @test InverseCount([89, 37, 49, 62, 87, 28, 72, 55, 19, 25, 48, 99, 51, 39, 38, 20, 61, 92, 48, 78]) == 9

  @test InverseCount([10, 35, 25, 74, 76, 35, 91, 23, 59, 28, 7, 90, 11, 15, 35, 54, 12, 80, 65, 78]) == 8

end

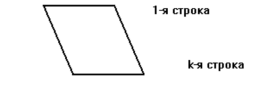
# Схема алгоритма

# Тестирование алгоритма

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование проверки** | **Ввод** | **Полученный результат** | **Вывод** |
| Тест 1 | 1 2 3 4 5 6 7 8 9 0 9 8 7 6 5 4 3 2 1 0 | 10 | Программа работает. |
| Тест 2 | 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 | 0 | Программа работает. |
| Тест 3 | 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 | 19 | Программа работает. |

# Задание 2

Дан трехмерный массив символов L(8,6,4). Вывести его элементы на экран строками по 12 символов в виде параллелограмма:



# Исходный код

function ParallelogramOutput(Array)

  c = 0

  for i in range(1, size(Array)[1])

    for j in range(1, size(Array)[2])

      for k in range(1, size(Array)[3])

        print(Array[i, j, k], " ")

      end

      if j % 3 == 0

        c += 1

        print("\n", " "^c)

      end

    end

  end

end

alph = "QWERTYUIOPASDFGHJKLZXCVBNM1234567890"

L = [alph[rand(1:36)] for x=1:8, y=1:6, z=1:4]

ParallelogramOutput(L)

# Схема алгоритма

# 

# Тестирование алгоритма

|  |  |  |
| --- | --- | --- |
| **Наименование проверки** | **Полученный результат** | **Вывод** |
| Тест 1 | U K N Y V R D B 9 J 5 H  3 L 7 7 Y K L U H G S 8  D X 8 5 Z O 4 2 C 4 M X  0 D J X E G Q G 2 M 4 V  R T 6 F R 4 V R D 8 Z N  5 5 W E Q R Z Z O Q U 2  1 X M X R K 8 N D 9 K M  W O B 8 V H B U M M C I  L A D 6 I X H T W H R 8  2 3 J S W O V S D R V 9  P 8 2 W F M T V I 1 8 2  C E B D 7 F H D D 9 S I  K 2 M 7 1 S Z D Y T T G  O 9 H 0 E 7 P X C B F D  9 6 L S 5 B 4 T X A 3 4  3 N 3 2 U 4 O R K 0 Z 5 | Программа работает |
| Тест 2 | U W J 0 A G Y 2 L K N V  3 G I Y V D J V V 4 R A  4 9 W 8 2 A 4 7 2 W 2 R  E B R H V 9 B 3 I M F V  R O H C E I S I 2 O X W  N E 6 H B K V A K 1 K R  N E W D T E 1 M 2 7 H I  R N 5 R S 8 V G Q W G 1  K R E B 9 D D V L 1 E 1  I M G 0 D U G P B 2 D B  7 V 1 Z N 5 Z H P 9 W 4  O Y R J P 2 K L M O 6 5  B G E 2 S P L V D H G H  K G F P G O 2 V 9 8 3 J  2 W J D H 8 C I A W 7 7  N V 4 L R J K 7 L I M 0 | Программа работает |

# Задание 3

Дано натуральное число n, латинские символы S1, S2,...,Sn, образующие строку. Группы символов, разделенных одним или несколькими пробелами и не содержащих пробелов внутри себя, назовем словами. Найти количество слов, начинающихся с буквы «A» и оканчивающиеся на «E».

# Исходный код

function IsFormatedWord(word, first = "", last = "")

  if in(word[1], first) && in(word[length(word)], last)

    return true

  else

    return false

  end

end

print("Введите количество слов в вашей строке: ")

n = parse(Int64, readline())

print("Введите вашу строку из ", n, " слов: ")

string\_ = split(readline())

if length(string\_) != n

  println("Количество слов не совпадает с введенным!")

end

words = []

for i in string\_

  if IsFormatedWord(i, "A", "E")

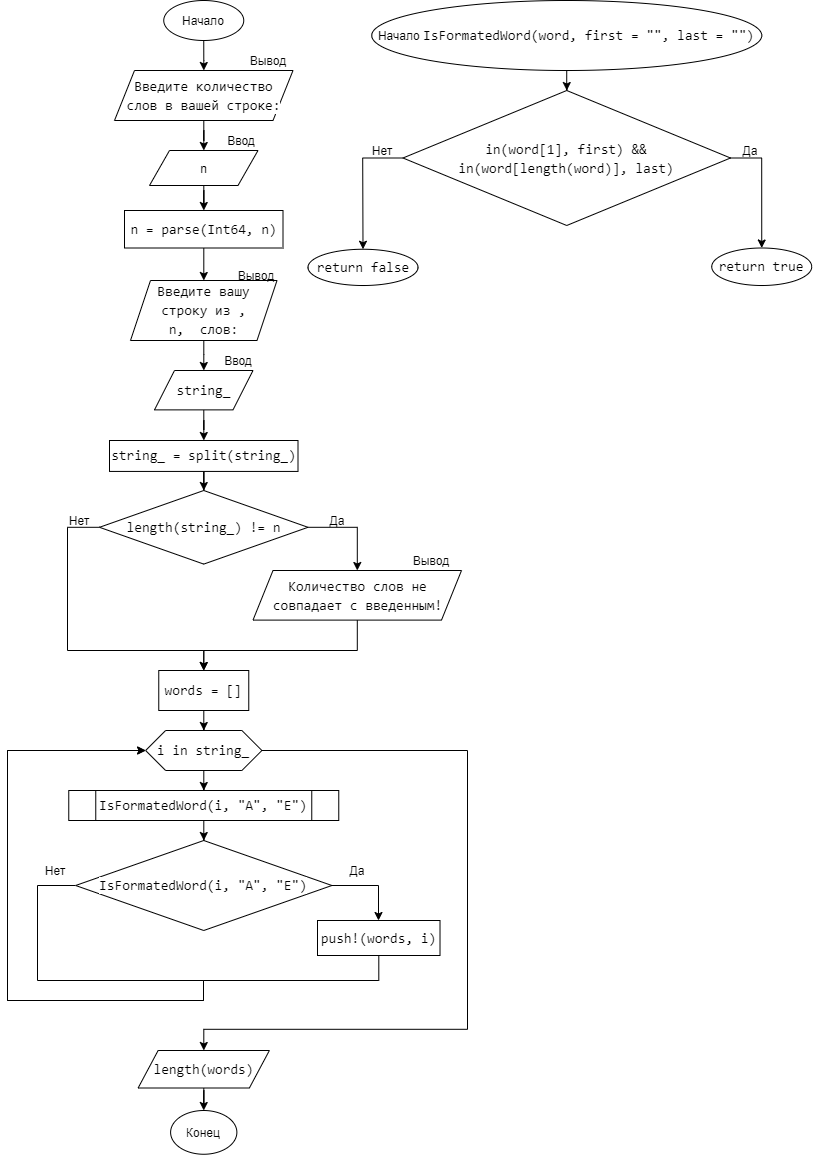
    push!(words, i)

  end

end

print(length(words))

# Схема алгоритма



# Тестирование алгоритма

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование проверки** | **Ввод** | **Полученный результат** | **Вывод** |
| Тест 1 | 5, Ae AE aE ae ABVGDE | 2 | Программа работает. |
| Тест 2 | 3, aaaaaa rrrrrr AE | 1 | Программа работает. |
| Тест 3 | 10, aa bb vv gg dd ee jj ii ss ff | 0 | Программа работает. |
| Тест 4 | 3, aa | Количество слов не совпадает с введенным!  0 | Программа работает. |

# Выводы

Выполняя лабораторную работу, познакомился с одномерными и двумерными массивами, научился работать с функциями в Julia.