



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)

**О Т Ч Е Т**

по лабораторной работе № 2

Название: Коллекции и строки

Дисциплина: Алгоритмизация и программирование

Студент

БМТ1-13Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата)

Д.А. Силантьев  
(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

Т.А.Ким  
(И.О. Фамилия)

Москва, 2022

# Задание 1

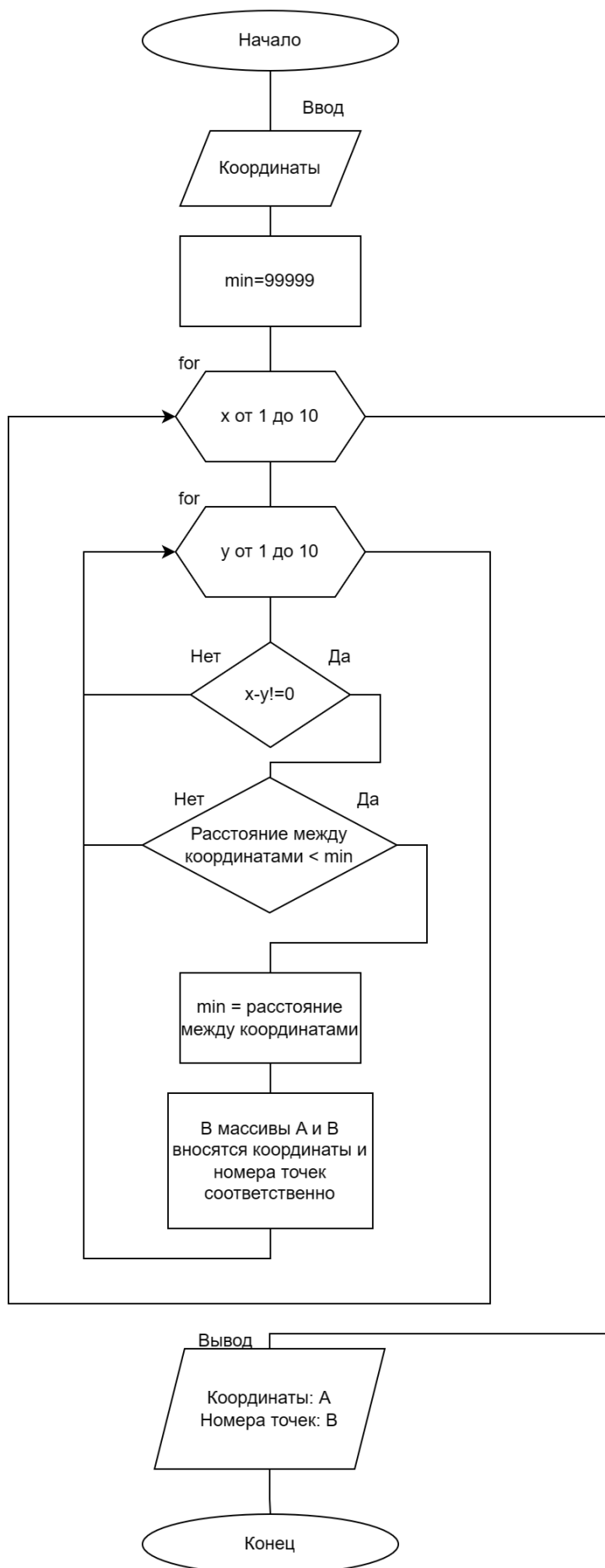
## Задание 1. Одномерные массивы

Даны координаты 10 точек на прямой. Определить номера двух точек, расстояние между которыми наименьшее (считать, что такая пара точек единственная). Вывести на экран массив, координаты и номера найденных точек.

## Исходный код

```
1  function main()
2  println("введите 10 координат точек")
3  a=parse(BigFloat, readline())
4  b=parse(BigFloat, readline())
5  c=parse(BigFloat, readline())
6  d=parse(BigFloat, readline())
7  e=parse(BigFloat, readline())
8  f=parse(BigFloat, readline())
9  g=parse(BigFloat, readline())
10 h=parse(BigFloat, readline())
11 i=parse(BigFloat, readline())
12 j=parse(BigFloat, readline())
13 min=99999
14 Array_T=[a,b,c,d,e,f,g,h,i,j]
15 Array_N=[a,b,c,d,e,f,g,h,i,j]
16 Array_A=[]
17 Array_B=[]
18 for x in range(1,10)
19     for y in range(1,10)
20         if x-y != 0
21             if (max((Array_T[x]-Array_N[y]), (Array_N[y]-Array_T[x]))<min)
22                 min=max((Array_T[x]-Array_N[y]), (Array_N[y]-Array_T[x]))
23                 Array_A=[Array_T[x],Array_N[y]]
24                 Array_B=[x,y]
25             end
26         end
27     end
28 end
29 println("Координаты:", Array_A)
30 println("Номера точек:", Array_B)
31 end
32
33 main()
```

# Схема алгоритма



## Тестирование алгоритма

Приводим результаты тестирования. Описываем сценарии проверок, ожидаемый результат и полученный результат. Делаем выводы...

Наименование проверки	Ожидаемый результат	Полученный результат	Вывод
Вводим 10 одинаковых координат	Выведет координаты первой и второй точки как минимальное расстояние	Вывел координаты первой и второй точки как минимальное расстояние	Программа работает

## Выводы

Я научился работать с одномерными массивами.

## Задание 2

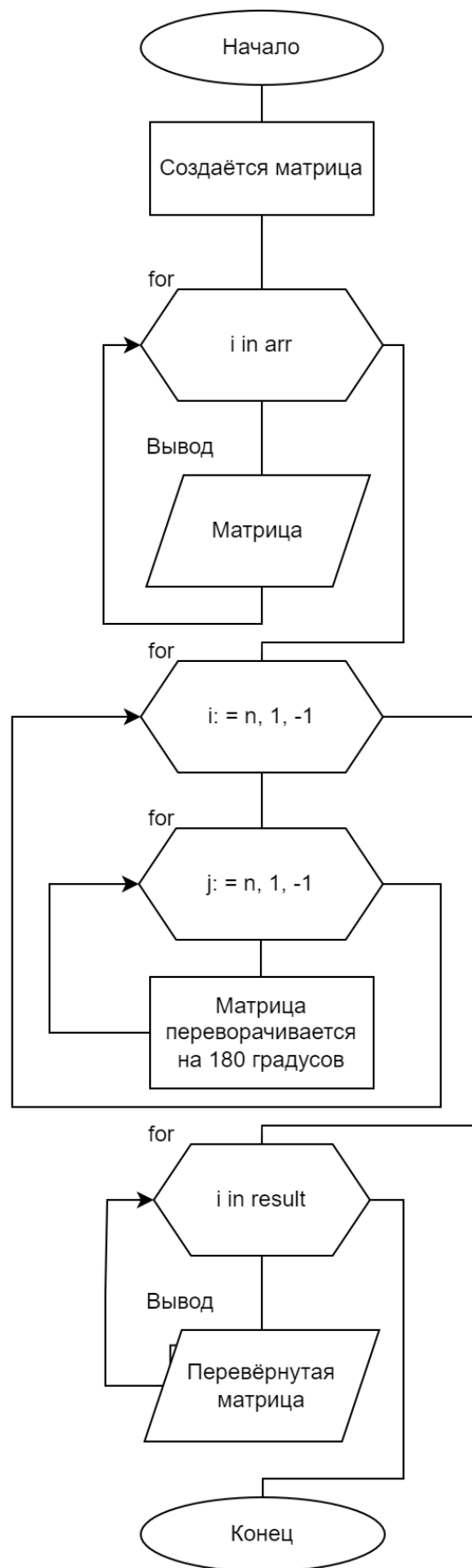
### Задание 2. Матрицы

Дана символьная матрица  $M(n, n), n \leq 10$ . Вывести на экран исходную матрицу, а затем эту же матрицу, повернув ее на  $180^\circ$ .

## Исходный код

```
1  n = rand(2:10)
2  arr = [rand(10:99, n) for i in 1:n]
3  for i in arr
4      println(i)
5  end
6  result = []
7
8  for i in range(n, 1, step=-1)
9      global result
10     l = []
11     for j in range(n, 1, step=-1)
12         push!(l, arr[i][j])
13     end
14     push!(result, l)
15 end
16 for i in result
17     println(i)
18 end
```

## Схема алгоритма



# Тестирование алгоритма

Приводим результаты тестирования. Описываем сценарии проверок, ожидаемый результат и полученный результат. Делаем выводы...

Наименование проверки	Ввод данных	Ожидаемый результат	Полученный результат	Вывод
Тест №1	[86, 99, 95] [83, 41, 90] [94, 97, 51]	[51, 97, 94] [90, 41, 83] [95, 99, 86]	[51, 97, 94] [90, 41, 83] [95, 99, 86]	Программа работает
Тест №2	[71, 27] [67, 28]	[28, 67] [27, 71]	[28, 67] [27, 71]	Программа работает

## Выводы

Я научился работать с матрицами.

## Задание 3

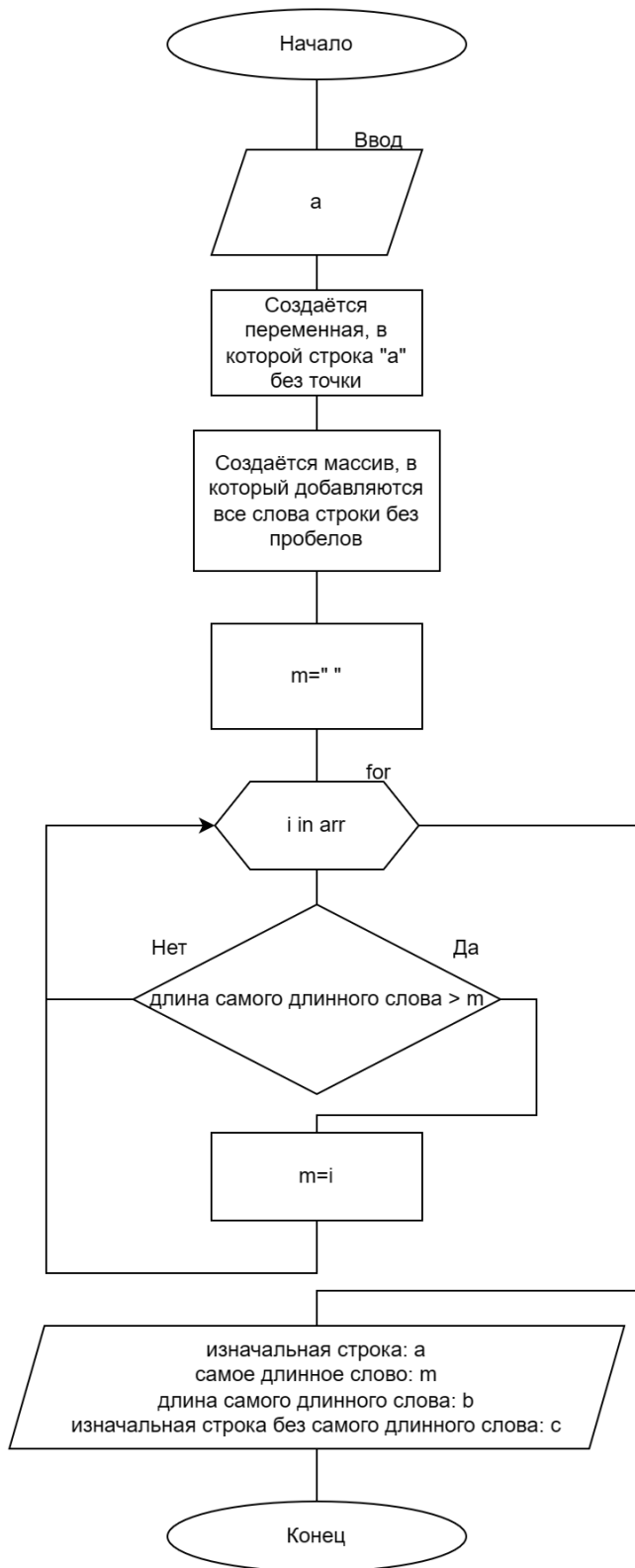
### Задание 3. Строки

Дана строка длиной не более 40 латинских символов. Последний символ – «.». Слова в тексте разделены несколькими пробелами. Удалить слово максимальной длины. Вывести на экран исходную строку, найденное слово, его длину и скорректированную строку.

## Исходный код

```
1  println("напечатайте строку")
2  a = readline()
3  d = a[1:end - 1]
4  arr = split(d)
5  m = ""
6  for i in arr
7      global m
8      if length(i) > length(m)
9          m = i
10     end
11 end
12 b=length(m)
13 c=replace(a, (" " * m) => "")
14 println("исходная строка:",a)
15 println("самое длинное слово:", m)
16 println("длина самого длинного слова:", b)
17 println("исходная строка без самого длинного слова:", c)
```

## Схема алгоритма





## Тестирование алгоритма

Приводим результаты тестирования. Описываем сценарии проверок, ожидаемый результат и полученный результат. Делаем выводы...

Наименование проверки	Ввод данных	Ожидаемый результат	Полученный результат	Вывод
Тест №1	wqfsfaewfsa gsdgsd gsajhhgffsj	изначальная строка:wqfsfaewfsa gsdgsd gsajhhgffsj самое длинное слово:wqfsfaewfsa длина самого длинного слова:11 изначальная строка без самого длинного слова:wqfsfaewfsa gsdgsd gsajhhgffsj	изначальная строка:wqfsfaewfsa gsdgsd gsajhhgffsj самое длинное слово:wqfsfaewfsa длина самого длинного слова:11 изначальная строка без самого длинного слова:wqfsfaewfsa gsdgsd gsajhhgffsj	Программа работает
Тест №2	fsd fds fdsa	изначальная строка:fsd fds fdsa самое длинное слово:fsd длина самого длинного слова:3 изначальная строка без самого длинного слова:fsd fds fdsa	изначальная строка:fsd fds fdsa самое длинное слово:fsd длина самого длинного слова:3 изначальная строка без самого длинного слова:fsd fds fdsa	Программа работает

## Выводы

Я научился работать со строками.