

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Рязанский государственный радиотехнический университет
имени В. Ф. Уткина»

Кафедра «Вычислительная и прикладная математика»

Отчет
по лабораторной работе № 3
по дисциплине
«Низкоуровневое программирование»
на тему
«Массивы и функции языка С»

Выполнил:
студент гр. 143
Вербицкая И. С.

Проверил:
Щенева Ю.Б.

Рязань 2022

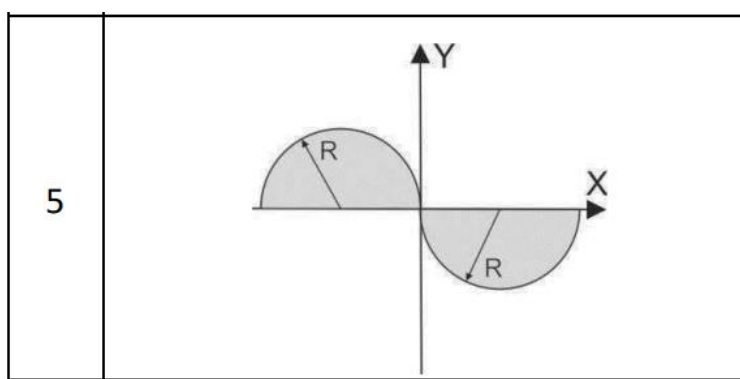
Задание (вариант №5):

Дан массив координат, необходимо определить какие из этих координат попадают в заданную закрашенную область, а какие не попадают, и вывести эту информацию на экран. Массив координат задаётся двумерным массивом, где первый индекс означает номер координаты, а второй номер компоненты координаты ($x - 0$, $y - 1$). Массив задаётся по выбору пользователя либо через непосредственный ввод координат с клавиатуры, либо случайным образом в заданном пользователем диапазоне.

В программе должно быть минимум четыре отдельных функции:

1. Задание массива координат через непосредственный ввод координат с клавиатуры.
2. Задание массива координат случайным образом в заданном диапазоне.
3. Определение попадания точки в заданную область.
4. Вывод попавших и не попавших координат в область.

Параметры **R**, **a** и **b**, задающие область, а также количество координат задаются пользователем.



Анализ задания:

Этапы решения задачи:

1. Составить блок-схему;
2. Составить программу;
3. Провести проверку работы программы.

Блок-схема:

На рисунке 1 изображена блок-схема подпрограммы заполнения массива координат случайными числами, на рисунке 2 – заполнения вручную.

На рисунке 3 изображена блок-схема функции, возвращающей 0 в случае, когда заданная точка не попадает в область, и 1 – когда попадает.

На рисунке 4 изображена блок-схема подпрограммы вывода попавших и не попавших в область координат.

На рисунке 5 изображена блок-схема основной программы.

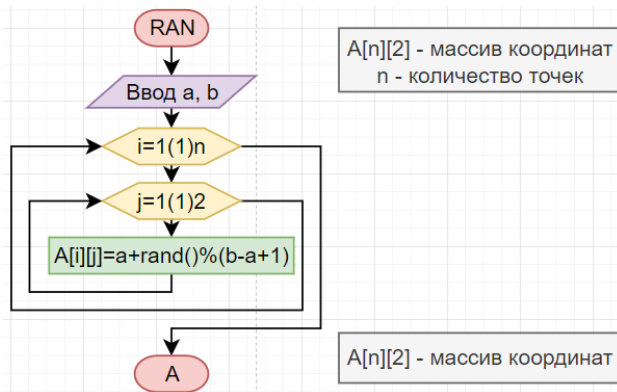


Рисунок 1

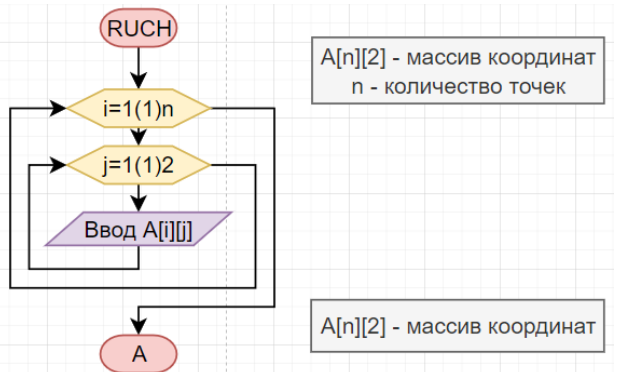


Рисунок 2

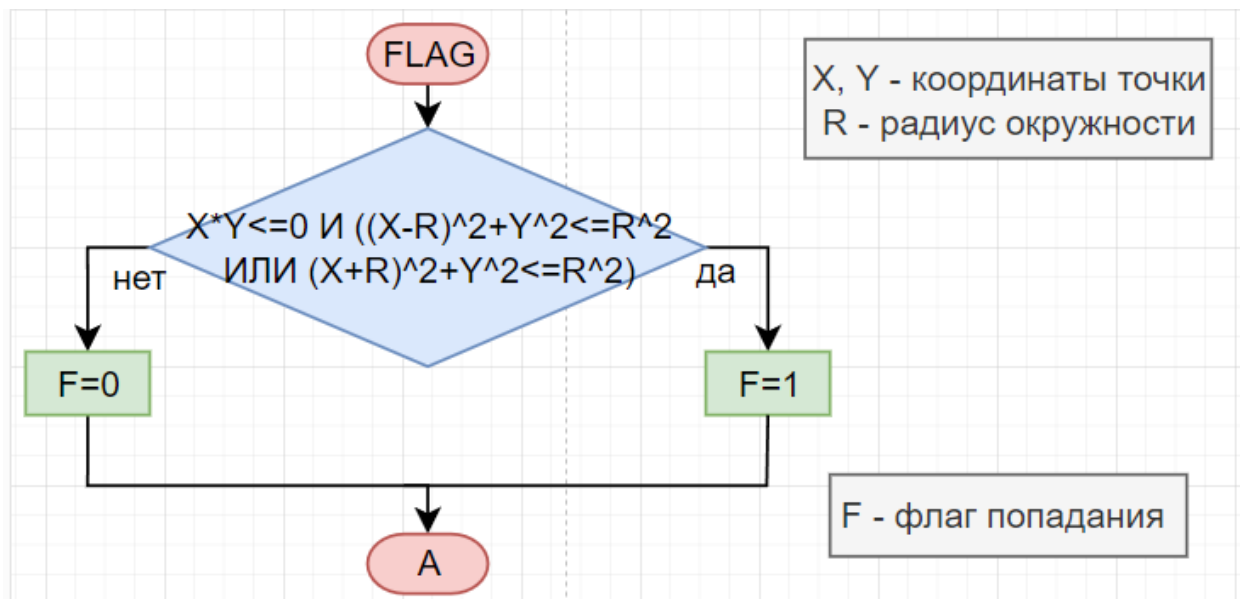


Рисунок 3

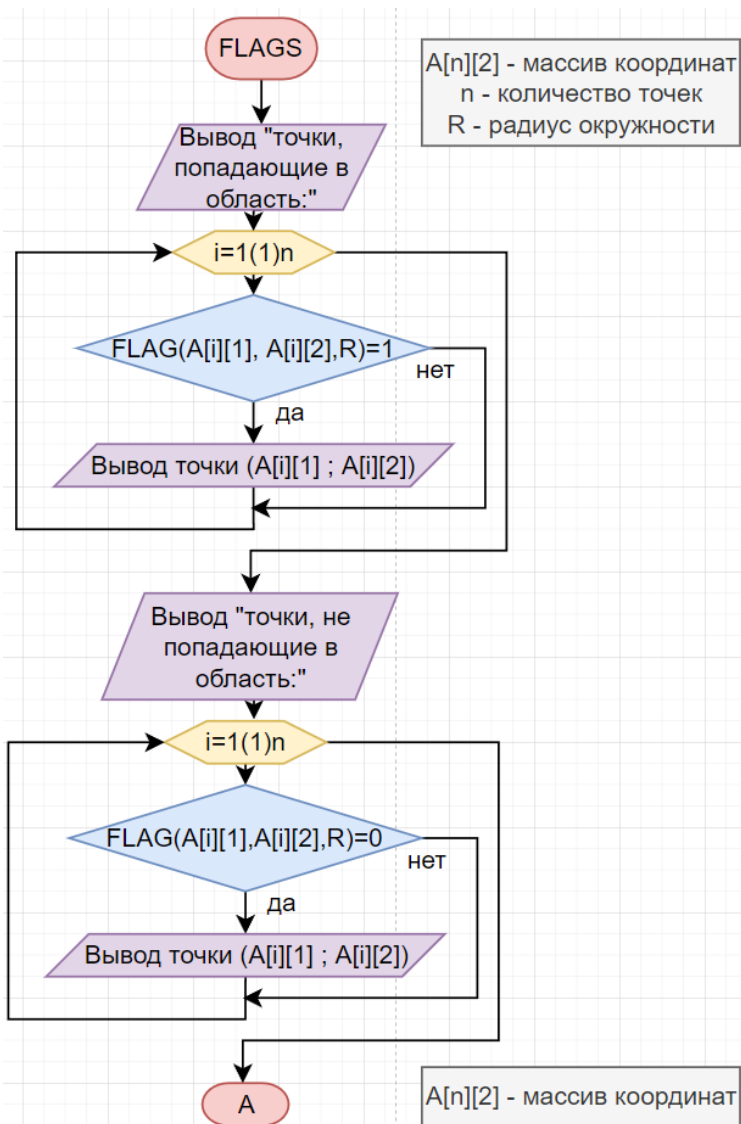


Рисунок 4

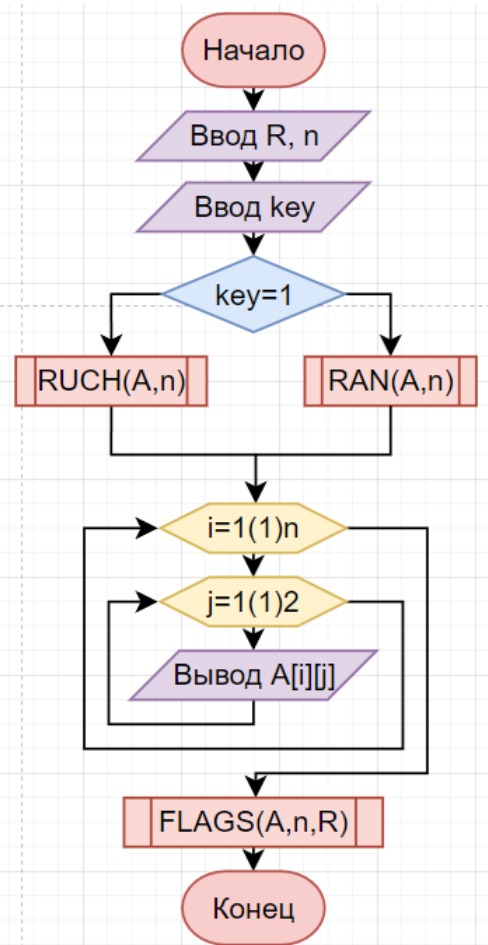


Рисунок 5

Листинг программы (рисунок 6, рисунок 7):

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4  #include <time.h>
5  /* x y
6     x y
7     x y
8     номера точек - строки i, значения x, y - столбцы j */
9
10 //ФУНКЦИЯ СЛУЧАЙНОГО ЗАПОЛНЕНИЯ ЗНАЧЕНИЙ КООРДИНАТ
11 void RAN (float A [1000][2], int n)
12 {
13     int a,b;
14     printf("Введите нижнюю границу диапазона случайных чисел, a=");
15     scanf("%d",&a);
16     printf("Введите верхнюю границу диапазона случайных чисел, b=");
17     do {
18         scanf("%d",&b);
19         if (b<=a)
20             printf("Ошибка: значение конца отрезка должно быть больше значения начала, b=");
21         while (b<=a);
22     } while (b<=a);
23     srand(time(NULL));
24     int i,j;
25     for (i=1;i<=n;i++)
26         for (j=1;j<=2;j++)
27             A[i][j]=a+rand()%(b-a+1);
28 }
  
```

Рисунок 6

```

28
29 //ФУНКЦИЯ РУЧНОГО ЗАПОЛНЕНИЯ ЗНАЧЕНИЙ X И Y
30 void RUCH (float A [1000][2], int n)
31 {
32     int i,j;
33     for (i=1;i<=n;i++)
34         for (j=1;j<=2;j++)
35         {
36             //printf("A[%d][%d]=",i,j);
37             if (j==1) printf("X %d=",i);
38             else printf("Y %d=",i);
39             scanf("%f",&A[i][j]);
40         }
41 }
42
43 //ФУНКЦИЯ ОПРЕДЕЛЕНИЯ ПОПАДАНИЯ КОНКРЕТНОЙ ТОЧКИ В ЗАДАННУЮ ОБЛАСТЬ
44 float FLAG(float X, float Y, float R)
45 {
46     float F;
47     if ((X*Y<=0) && ((powf(X-R,2)+Y*Y<=R^2) || (powf(X+R,2)+Y*Y<=R^2)))
48         F=1;
49     else F=0;
50     return F;
51 }
52
53 //ФУНКЦИЯ ВЫВОДА ПОПАВШИХ И НЕ ПОПАВШИХ В ОБЛАСТЬ КООРДИНАТ
54 void FLAGS(float A [1000][2], int n, float R)
55 {
56     printf("\nТОЧКИ, ПОПАДАЮЩИЕ В ОБЛАСТЬ:\n");
57     int i;
58     for(i=1;i<=n;i++)
59         if (FLAG(A[i][1],A[i][2],R)==1)
60             printf("(%.1f ; %.1f)\n",A[i][1],A[i][2]);
61     printf("ТОЧКИ, НЕ ПОПАДАЮЩИЕ В ОБЛАСТЬ:\n");
62     for(i=1;i<=n;i++)
63         if (FLAG(A[i][1],A[i][2],R)==0)
64             printf("(%.1f ; %.1f)\n",A[i][1],A[i][2]);
65 }
66
67
68 //ОСНОВНАЯ ПРОГРАММА
69 int main()
70 {
71     system("chcp 1251");
72     float A [1000][2];
73     //задаем значение R
74     printf("Введите радиус окружностей, R=");
75     float R;
76     do {
77         scanf("%f",&R);
78         if (R<=0)
79             printf("Ошибка: радиус должен быть больше нуля, R=");
80     } while (R<=0);
81     //ввод количества координат
82     printf("Введите количество точек, n=");
83     int n;
84     do {
85         scanf("%d",&n);
86         if (n<=0)
87             printf("Ошибка: количество должно быть больше нуля, n=");
88         if (n>1000)
89             printf("Ошибка: количество не может быть больше тысячи, n=");
90     } while ((n>1000) || (n<=0));
91     //выбор и реализация способа заполнения массива
92     int key;
93     printf("Введите 1 - для случайных значений x и y;\nлюбое другое число - ручной ввод значений x и y:\n");
94     scanf("%d",&key);
95     if (key==1) RAN(A,n); else RUCH(A,n);
96     //выводим все точки
97     printf("ИСХОДНЫЕ ЗНАЧЕНИЯ:\n");
98     printf("      X      Y\n");
99     int i,j;
100     for (i=1;i<=n;i++)
101     {for (j=1;j<=2;j++)
102     printf("%.1f ",10,A[i][j]);
103     printf("\n");
104     }
105     //выводим попавшие и не попавшие точки
106     FLAGS(A,n,R);
107 }

```

Рисунок 7

Результаты работы программы и проверка:

Программа была запущена для значений R=10, n=5, x1=1, y1=1, x2=0, y2=0, x3=-5, y3=5, x4=5, y4=-5, x5=5, y5=5 (рисунок 8). Результаты, полученные с помощью программы, совпали с результатами, полученными с помощью Excel (рисунок 9).

```
Текущая кодовая страница: 1251
Введите радиус окружностей, R=10
Введите количество точек, n=5
Введите 1 - для случайных значений x и y;
любое другое число - ручной ввод значений x и y:
2
X 1=1
Y 1=1
X 2=0
Y 2=0
X 3=-5
Y 3=5
X 4=5
Y 4=-5
X 5=5
Y 5=5

ИСХОДНЫЕ ЗНАЧЕНИЯ:
      X      Y
    1.0    1.0
    0.0    0.0
   -5.0    5.0
    5.0   -5.0
    5.0    5.0

ТОЧКИ, ПОПАДАЮЩИЕ В ОБЛАСТЬ:
(0.0 ; 0.0)
(-5.0 ; 5.0)
(5.0 ; -5.0)
ТОЧКИ, НЕ ПОПАДАЮЩИЕ В ОБЛАСТЬ:
(1.0 ; 1.0)
(5.0 ; 5.0)

-----
Process exited after 21.05 seconds with return value 6
Для продолжения нажмите любую клавишу . . .
```

Рисунок 8

| | A | B | C | D |
|---|----|----|----|-----------|
| 1 | R | x | y | Попадание |
| 2 | 10 | 1 | 1 | нет |
| 3 | | 0 | 0 | да |
| 4 | | -5 | 5 | да |
| 5 | | 5 | -5 | да |
| 6 | | 5 | 5 | нет |

Рисунок 9