Министерство науки и образования РФ

Федеральное государственное автономное образовательное

учреждение высшего профессионального образования

«Санкт-Петербургский государственный электротехнический

университет «ЛЭТИ» им. В. И. Ульянова (Ленина)»

(СПбГЭТУ «ЛЭТИ»)

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

Отчет по лабораторной работе №1

на тему:

“Условные операторы на языке С/C++”

по дисциплине “Программирование”

Выполнил: Табаков А.В.

Группа 4306

Принял: к.т.н., доцент Сискович Т.И.

Санкт-Петербург  
2014 г.

**Цель**

Получить практические навыки в разработке разветвляющихся алгоритмов и использовании условных операторов на языке «С».

**Задание**

Разработать алгоритм и написать программу, проверяющую многократно, по желанию пользователя, попадает ли точка с заданными координатами (X;Y) в указанную на рис.1 область, находится на границе области или вне области.

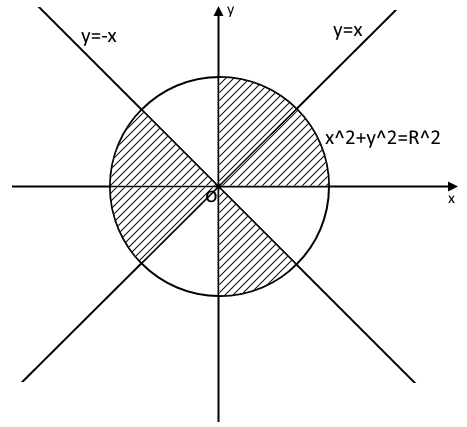
****

Рис.1. Заданные области

**Уточнение задания**

Вначале программа должна предлагать пользователю задать радиус окружности и две координаты точки x и y на плоскости.

При этом программа должна проверить вводимые пользователем значения. Радиус должен быть положительным. После вывода результата программа должна предложить пользователю ввести новые значения координат точки и/или радиус с помощью переменных, значения которых вводится с клавиатуры и соответствующих организации циклов.

**Контрольные примеры**

Контрольные примеры представлены в Таблице 1*.*

Таблица 1. Контрольные примеры

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Исходные данные | | | Результат |
| R | X | Y |
| 1 | -1 | - | - | Радиус должен быть положительным. Введите радиус! |
| 2 | 0 | - | - | Радиус должен быть положительным. Введите радиус! |
| 3 | 5 | 0 | 0 | Точка лежит в начале осей координат |
| 4 | 5 | 3 | 0 | Точка лежит на границе |
| 5 | 5 | 1 | 2 | Точка принадлежит данной области |
| 6 | 5 | -3 | -1 | Точка принадлежит данной области |
| 7 | 5 | 4 | 3 | Точка лежит на границе |
| 8 | 5 | 0 | 3 | Точка лежит на границе |
| 9 | 5 | 1 | 1 | Точка лежит на границе |
| 10 | 5 | -6 | -3 | Точка не принадлежит данным областям |
| 11 | 5 | -1 | -4 | Точка не принадлежит данным областям |
| 12 | 5 | -1 | 4 | Точка не принадлежит данным областям |

**Описание переменных**

Описание переменныхпредставлено вТаблице 2.

Таблица 2*.*Описание переменных

|  |  |  |
| --- | --- | --- |
| **Имя переменной** | **Тип** | **Назначение** |
| R | int | Радиус окружности |
| Х | int | Координата x точки |
| Y | int | Координата y точки |
| Q | char | Вспомогательная переменная для организации цикла |

**Описание алгоритма**

Шаг № 1

Вывод сообщения на экран для пользователя с просьбой ввести значение радиуса.

Шаг № 2

Ввод пользователем радиуса.

Шаг № 3

Проверка значения, введенное пользователем. Значение R должно быть положительным, т.к. иначе не будет областей для проверки, если это условие не соблюдено, то – вывод сообщения об ошибке и возврат к шагу № 2,в противном случае переход к шагу № 4.

Шаг № 4

Вывод сообщения на экран для пользователя с просьбой ввести значение X,Y.

Шаг № 5

Ввод пользователем X,Y.

Шаг № 6

Если x^2+y^2>R^2, то вывод сообщения: «Точка не принадлежит данным областям» и переход к шагу №, иначе переход к шагу № 7.

Шаг № 7

Если x^2+y^2=R^2, то вывод сообщения: «Точка лежит на границе» и переход к шагу №, иначе переход к шагу № 8.

Шаг № 8

Если X=0 и Y=0, то вывод сообщения: «Точка лежит на начале координат» и переход к шагу №, иначе переход к шагу № 9.

Шаг № 9

Если X=0 и Y>0 или X>0 и Y=0, то вывод сообщения: «Точка лежит на границе» и переход к шагу №, иначе переход к шагу № 10.

Шаг № 10

Если X>=0, то переход к шагу № 11, иначе переход к шагу № 13.

Шаг № 11

Если Y>0 и Y=X или Y<0 и Y<-X и X!=0, то вывод сообщения: «Точка не принадлежит данным областям» и переход к шагу № 16, иначе переход к шагу № 12.

Шаг № 12

Если Y>-X и Y!=X, то вывод сообщения: «Точка не принадлежит данным областям» и переход к шагу №16, иначе вывод сообщения: «Точка принадлежит данным областям» переход к шагу № 16.

Шаг № 13

Если Y<-X и Y>X и Y!=0, то вывод сообщения: «Точка принадлежит данным областям» и переход к шагу № 16, иначе переход к шагу № 14.

Шаг № 14

Если Y<X или Y>-X, то вывод сообщения: «Точка не принадлежит данным областям» и переход к шагу № 16, иначе переход к шагу № 15.

Шаг № 15

Если Y=X или Y=-X и Y=0, то вывод сообщения: «Точка лежит на границе» и переход к шагу №16, иначе вывод сообщения: «Точка не принадлежит данным областям» переход к шагу № 16.

Шаг № 16

Вывод сообщения: «Проверить другую точку при заданном радиусе? Для повтора введите 'Y'».

Шаг № 17

Если на вопрос пользователь отвечает ‘Y’ или ‘y’, то осуществляется переход к шагу № 4.

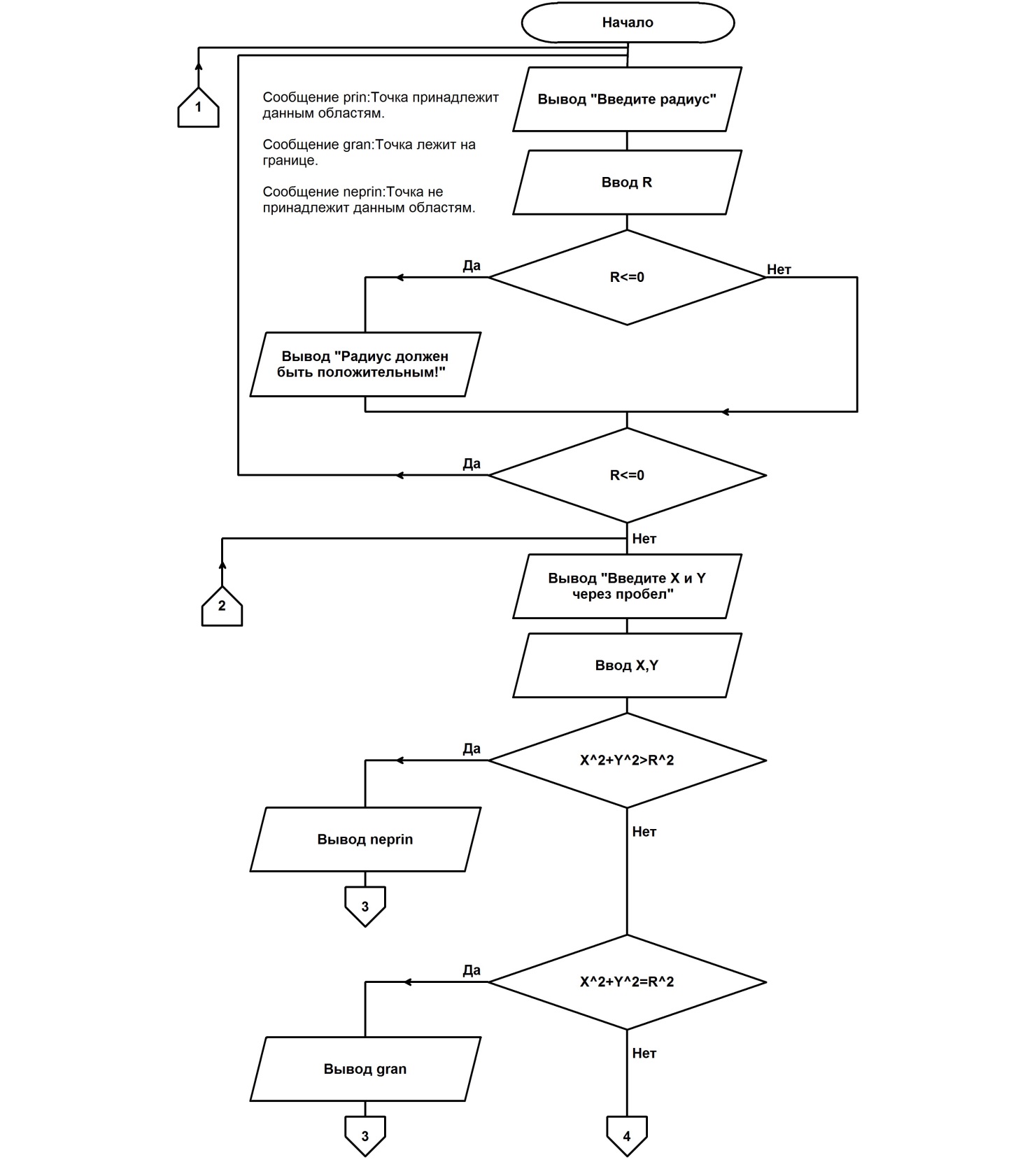
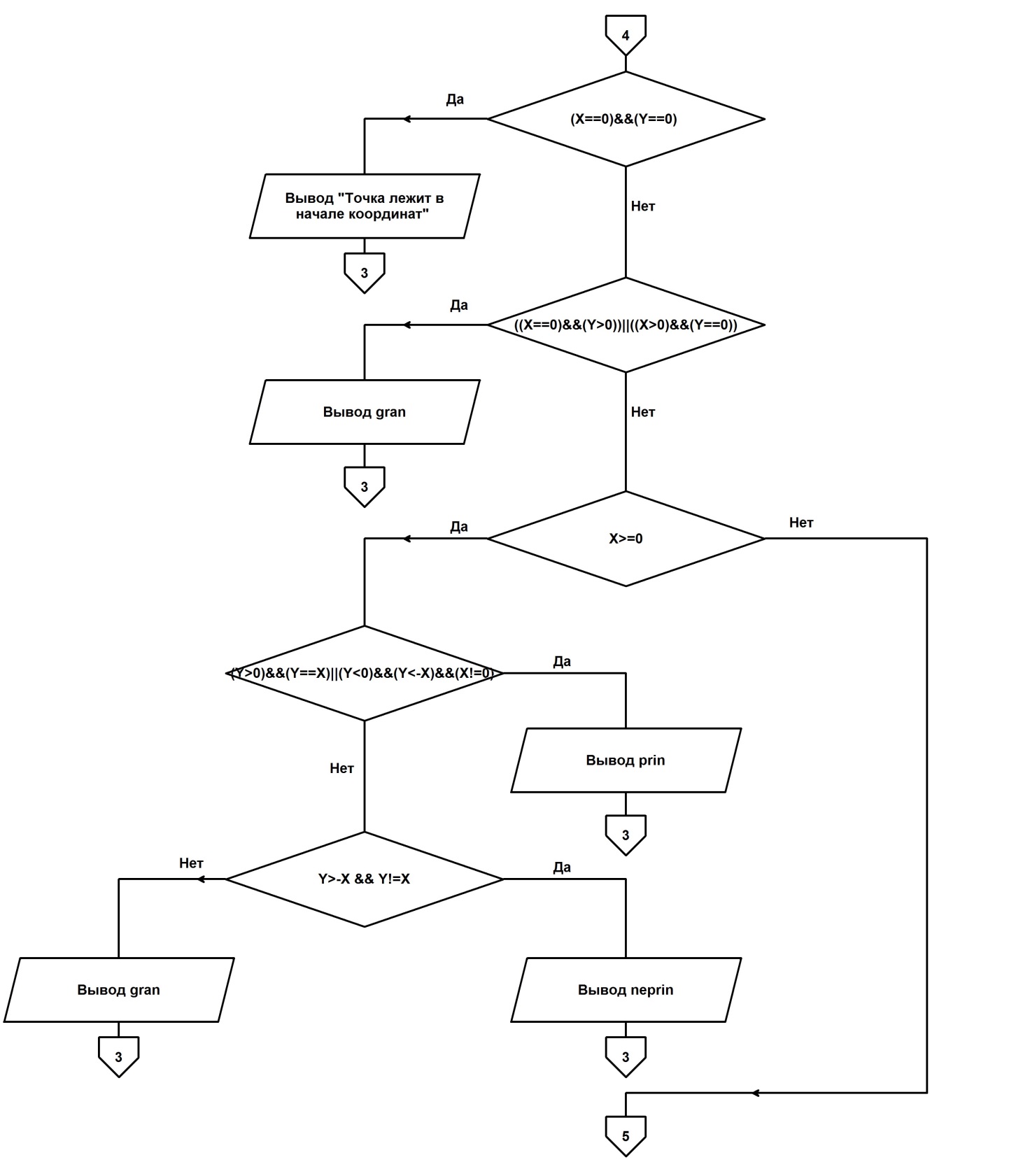
Шаг № 18

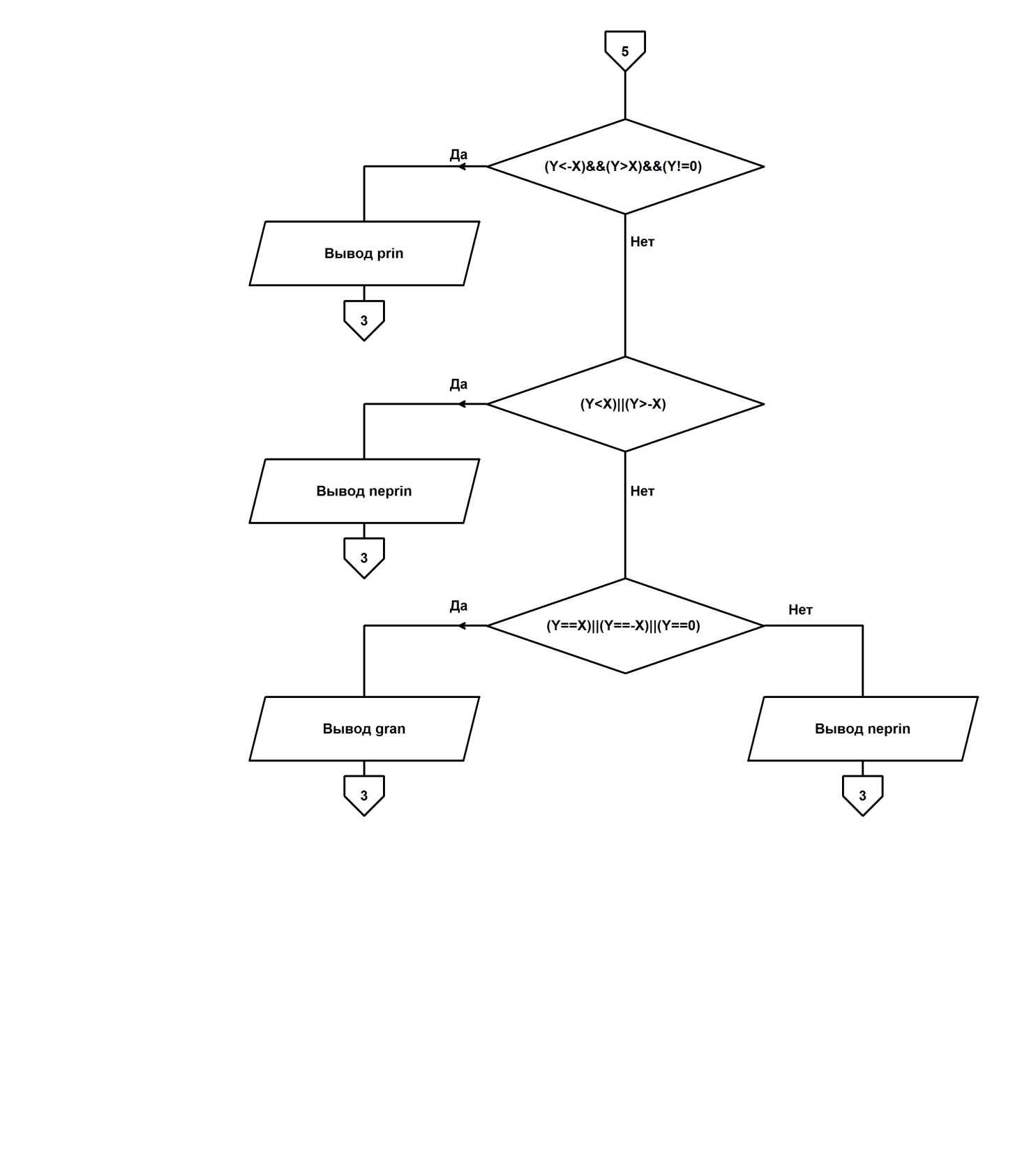
Вывод сообщения: «Повторить программу заново? Для повтора введите 'Y'».

Шаг № 19

Если на вопрос пользователь отвечает ‘Y’ или ‘y’, то осуществляется переход к шагу № 2, иначе программа завершается.

**Блок-схема программы**

Блок-схема программы представлена на рисунках 2-5.** **



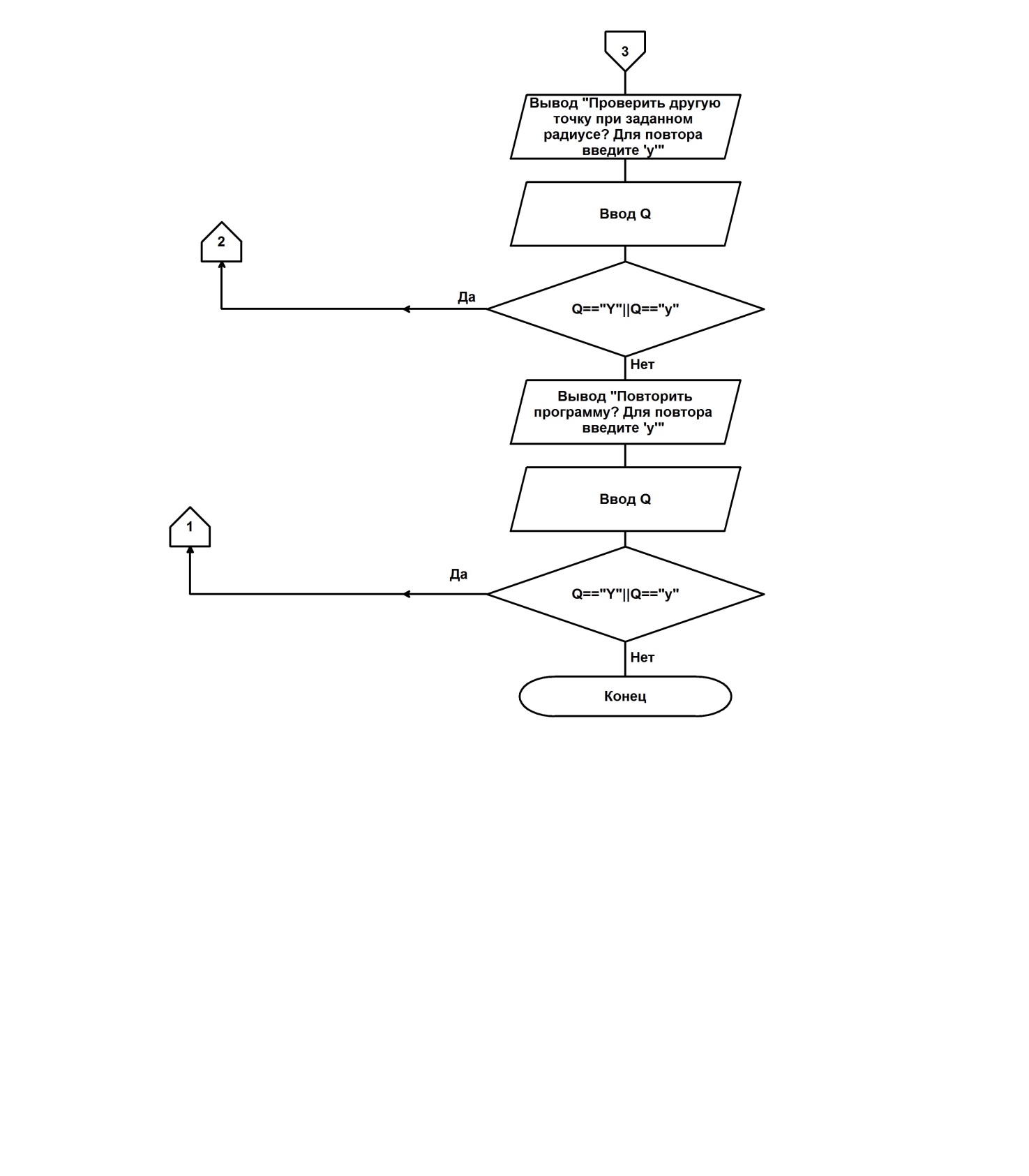


Рис.2. Блок-схема

**Текст программы с комментариями**

#include <stdio.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "RUS");

int X, Y, R;

char Q;

do

{

printf("Введите радиус\n");

do

{

scanf("%d", &R);

if (R <= 0)

printf("Радиус должен быть положительным! Введите радиус повторно!\n");

}

while (R <= 0);

do

{

printf("Введите координаты точки X и Y через пробел\n");

scanf("%d%d", &X, &Y);

if (X\*X+Y\*Y>R\*R) //принадлежит ли точка окружности

printf("Точка не принадлежит данным областям\n");

else

if (X\*X+Y\*Y==R\*R) //лежит ли точка на границе окружности

printf("Точка лежит на границе\n");

else

if ((X==0) && (Y==0))

printf("Точка лежит на начале координат\n");

else

if (X\*X+Y\*Y<R\*R) //лежит ли точка в окружности

{

if (((X==0) && (Y>0)) || ((X>0) && (Y==0)))

printf("Точка лежит на границе\n");

else

if (X>=0)

//проверка правой части координатной плоскости, относительно оси’y’

if (((Y>0) && (Y==X)) || ((Y<0) && (Y<-X) && (X!=0)))

printf("Точка принадлежит данным областям\n");

else

if (Y>-X && Y!=X)

printf("Точка не принадлежит данным областям\n");

else

printf("Точка лежит на границе\n");

else //проверка левой части координатной плоскости, относительно оси’y’

if ((Y<-X) && (Y>X) && (Y!=0))

printf("Точка принадлежит данным областям\n");

else

if ((Y<X) || (Y>-X))

printf("Точка не принадлежит данным областям\n");

else

if (((Y==X) || (Y==-X) || (Y==0)))

printf("Точка лежит на границе\n");

else

printf("Точка не принадлежит данным областям\n");

}

printf("Проверить другую точку при заданном радиусе? Для повтора введите 'Y'\n");

fflush(stdin);

scanf("%c", &Q);

}

while (Q=='Y' || Q=='y'); //цикл повторного ввода координат при заданном радиусе

printf("Повторить программу заново? Для повтора введите 'Y'\n");

fflush(stdin);

scanf("%c", &Q);

}

while (Q=='Y' || Q=='y'); //цикл повтора программы

return 0;

}

**Результаты решения задачи**

При выполнении программы были получены результаты, совпадающие со значениями, приведенными в Таблице 1.Ошибок не обнаружено.

**Вывод**

При выполнении лабораторной работы были получены практические навыки в разработке разветвляющихся алгоритмов и использовании условных операторов на языке программирования «С/C++».