ГУАП

КАФЕДРА № 42

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доцент |  |  |  | В. А. Кузнецов |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 1.1 |
| ПРОСТОЙ АЛГОРИТМ С ИСПОЛЬЗОВАНИЕМ УСЛОВНОГО ОПЕРАТОРА |
| по курсу: |
| ОСНОВЫ ПРОГРАММИРОВАНИЯ |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ гр. № | 4326 |  |  |  | Г. С. Томчук |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2024

СОДЕРЖАНИЕ

[1 Постановка задачи 3](#_Toc160063880)

[2 Схема алгоритма решения 4](#_Toc160063881)

[3 Полное описание реализованной функции 5](#_Toc160063882)

[4 Листинг программы 6](#_Toc160063883)

[5 Несколько тестов работы программы 7](#_Toc160063884)

1. Постановка задачи

Задача: реализовать программную функцию на языке C/С++, выполняющую поставленную задачу. Вариант задания, пример входных и выходных данных представлен в таблице 1. Глобальные параметры использовать запрещено; допустимо использование дополнительных функций.

Таблица 1 – Вариант

|  |  |  |  |
| --- | --- | --- | --- |
| N | Текст задания | Вход | Выход |
| 7 | Реализовать функцию, которая возвращает номер соответствующей области, в которой находится заданная точка P, согласно схеме ниже.    По заданным P, a, b, R. Центр круга в точке [0,0]. a > b. | 2,2  1  -1  3 | 1 |

1. Схема алгоритма решения

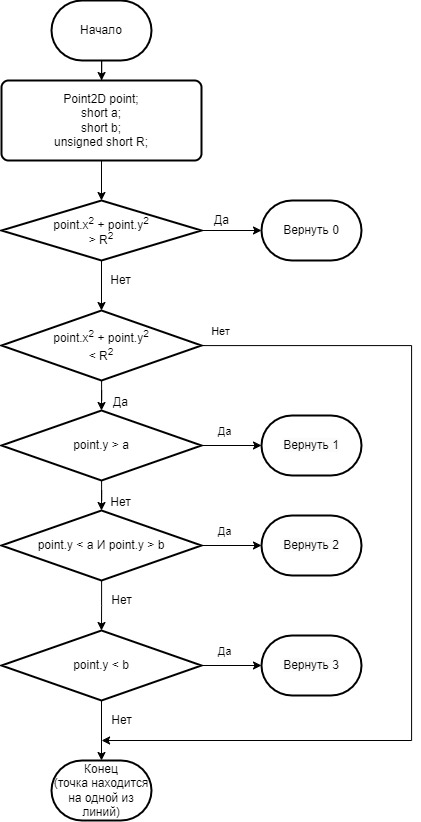


Рисунок 1 – Блок-схема алгоритма

1. Полное описание реализованной функции

Функция func принимает следующие аргументы:

1. Point2D point – структура, содержащая в себе два компонента (int x, int y), представляющая собой точку в двумерном пространстве.
2. short a – значение, задающее высоту прямой y=a.
3. short b – значение, задающее высоту прямой y=b.
4. unsigned short R – радиус окружности.

Возвращает функция число int от 0 до 3 включительно, в зависимости от того, в какой области согласно входным данным находится точка point.

Помимо основной функции func, в теле функции main была реализована проверка на ввод целочисленных аргументов a, b и R, а также проверка на условие a > b.

Присвоение значений x и y осуществляется путём ввода строки типа "число,число" и последующим разбиением строки на два числа через запятую:

1. char \*ptr = strtok(point\_str, ","); – strtok возвращает указатель типа char\* на первую найденную лексему в строке point\_str с разделением по запятой.
2. int x = strtol(ptr, nullptr, 10);

ptr = strtok(nullptr, ",");

int y = strtol(ptr, nullptr, 10);

strtol возвращает интерпретированное целочисленное значение x в строке байтов, на которую указывает ptr; после повторного присвоения ptr и обращения к strtok с указанием пустого указателя, strtol аналогично возвращает значение y (т.е. strtok продолжает обрабатывать оригинальную строку, если передать nullptr).

1. Листинг программы

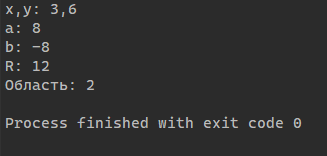
Листинг 1

#include <iostream>  
#include <cmath>  
#include <windows.h>  
#include <cstring>  
  
struct Point2D {  
 int x;  
 int y;  
};  
  
int func(Point2D point, short a, short b, unsigned short R);  
  
int invalid\_input();  
  
int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 char point\_str[32];  
 short a, b;  
 unsigned short R;  
  
 std::cout << "x,y: ";  
 std::cin >> point\_str;  
  
 char \*ptr = strtok(point\_str, ",");  
 int x = strtol(ptr, nullptr, 10);  
 ptr = strtok(nullptr, ",");  
 int y = strtol(ptr, nullptr, 10);  
  
 std::cout << "a: ";  
 std::cin >> a;  
 if (std::cin.fail()) {  
 invalid\_input();  
 }  
  
 std::cout << "b: ";  
 std::cin >> b;  
 if (std::cin.fail()) {  
 invalid\_input();  
 }  
  
 if (a <= b) {  
 std::cerr << "Должно выполнятся условие: a > b" << std::endl;  
 return 1;  
 }  
  
 std::cout << "R: ";  
 std::cin >> R;  
 if (std::cin.fail()) {  
 invalid\_input();  
 }

Продолжение листинга 1

int result = func({x, y}, a, b, R);  
  
 std::cout << "Область: " << result << std::endl;  
  
 return 0;  
}  
  
int func(Point2D point, short a, short b, unsigned short R) {  
 if (pow(point.x, 2) + pow(point.y, 2) > pow(R, 2)) {  
 return 0;  
 } else if (pow(point.x, 2) + pow(point.y, 2) < pow(R, 2)) {  
 if (point.y > a) {  
 return 1;  
 }  
 if (point.y < a && point.y > b) {  
 return 2;  
 }  
 if (point.y < b) {  
 return 3;  
 }  
 }  
 std::cout << "Точка находится на одной из линий" << std::endl;  
 exit(0);  
}  
  
int invalid\_input() {  
 std::cerr << "Введите целое число" << std::endl;  
 exit(1);  
}

1. Несколько тестов работы программы



Рисунок

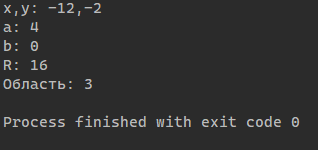


Рисунок 3

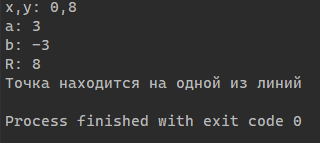


Рисунок 4

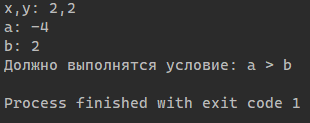


Рисунок 5

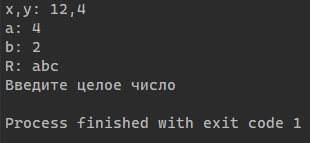


Рисунок 6