

ГУАП
КАФЕДРА №43

ОТЧЁТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

К.Т.Н., доц..
должность, уч. степень, звание

подпись, дата

А.В. Туманова
инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №2

ОПРЕДЕЛЕНИЕ ПОПАДАНИЯ ТОЧКИ В ОБЛАСТЬ

по дисциплине: ОСНОВЫ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. Z7431

 21.11.2018
подпись, дата

М.Д.Семочкин
инициалы, фамилия

Санкт-Петербург
2018

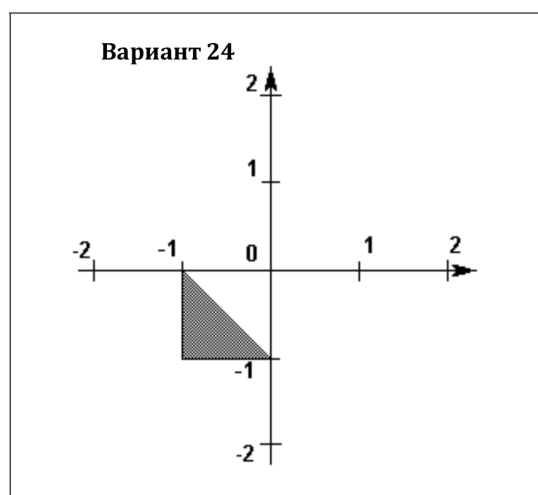
1. Цель работы

Целью работы является изучение логических операций типа НЕ, И, ИЛИ.

2. Задание

Согласно варианту №24,

Написать программу, которая для вводимых координат точек (x, y) определяет, попадает ли точка в заштрихованную область на рисунке, который соответствует индивидуальному варианту. Попадание на границу области считать попаданием в область.



Примечание: для проверки попадания точки в область следует использовать один условный оператор с несколькими условиями.

3. Описание созданных функций

Для реализации задания нам потребуются следующие функции:

Имя: `checkIfPointInZone`

Назначение: Определить, находится ли точка в заштрихованной области плоскости

Входные данные:

- x – значение x
- y – значение y

Выходные данные:

- `true` если точка находится в области, иначе `false`

Побочный эффект: отсутствует.

Тестовые данные:

x	y	результат
-0.6	-1.1	false
-1	-1	true
-0.6	-1	true
-0.6	-0.6	true
-0.5	-0.5	true
-0.5	-0.3	false
0	0	false
2	3	false

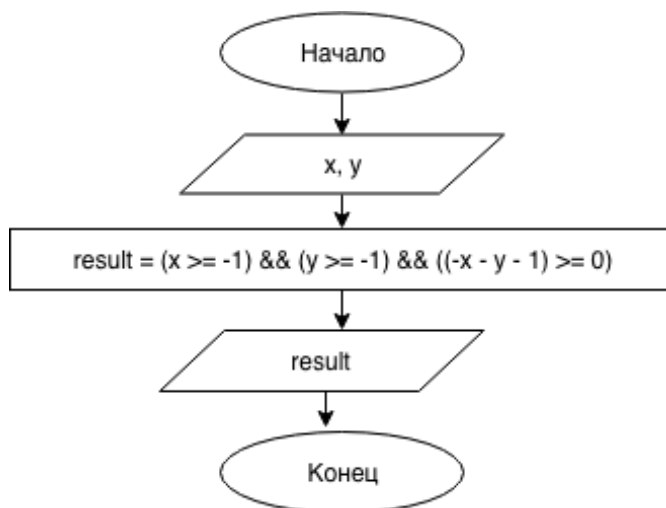
Прототип: `double checkIfPointInZone(const double x, const double y)`

Алгоритм:

- псевдокод

вернуть результат вычисления логического выражения “ $(x \geq -1) \text{ И } (y \geq -1) \text{ И } ((-x - y - 1) \geq 0)$ ”

- блок-схема



4. Листинг программы

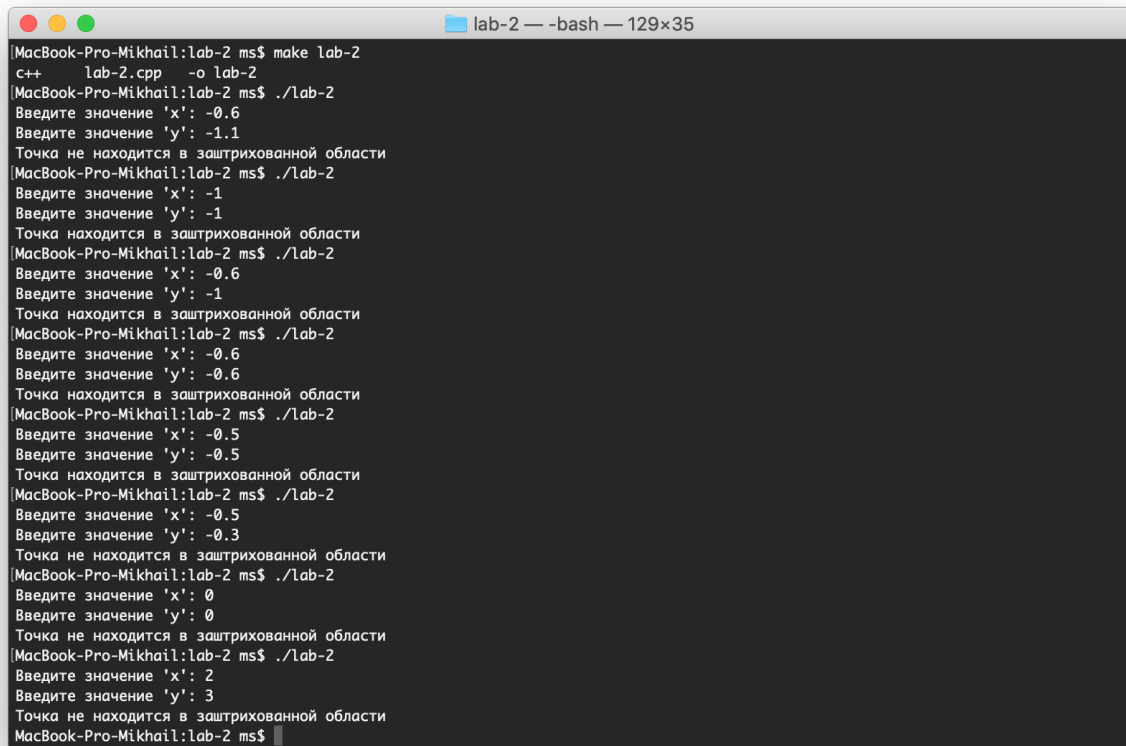
```
#include <iostream>
using namespace std;

double checkIfPointInZone(const double x, const double y) {
    // Уравнение прямой:  $-x - y - 1 = 0$ 
    return (x >= -1) && (y >= -1) && ((-x - y - 1) >= 0);
}

int main() {
    double x, y;
    bool isPointInZone;
    setlocale(LC_ALL, "russian");
    cout << "Введите значение 'x': ";
    cin >> x;
    cout << "Введите значение 'y': ";
    cin >> y;
    isPointInZone = checkIfPointInZone(x, y);
    if (isPointInZone) {
        cout << "Точка находится в заштрихованной области" << endl;
    } else {
        cout << "Точка не находится в заштрихованной области" << endl;
    }
    return 0;
}
```

5. Пример выполнения программы

Ниже показан пример выполнения программы.



```
[MacBook-Pro-Mikhail:lab-2 ms$ make lab-2
c++ lab-2.cpp -o lab-2
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -0.6
Введите значение 'y': -1.1
Точка не находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -1
Введите значение 'y': -1
Точка находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -0.6
Введите значение 'y': -1
Точка находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -0.6
Введите значение 'y': -0.6
Точка находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -0.5
Введите значение 'y': -0.5
Точка находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': -0.5
Введите значение 'y': -0.3
Точка не находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': 0
Введите значение 'y': 0
Точка не находится в заштрихованной области
[MacBook-Pro-Mikhail:lab-2 ms$ ./lab-2
Введите значение 'x': 2
Введите значение 'y': 3
Точка не находится в заштрихованной области
MacBook-Pro-Mikhail:lab-2 ms$
```

Видно, что результаты расчётов совпадают с тестовыми данными.

6. Анализ результатов и выводы

К достоинствам программы можно отнести:

- Программа выполняет поставленную задачу и работает без ошибок (для корректных тестовых данных).
- Задание реализовано в виде отдельной функции.

Из недостатков можно отметить:

- Не производится проверка входных данных.
- Функция проверки работает только для конкретного задания.