ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | М. А. Мурашова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

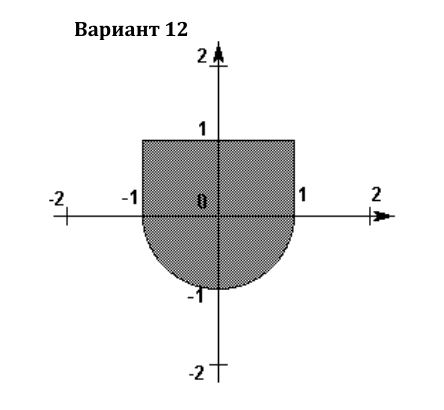
|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| Определение попадания точки в область |
| по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4135К |  |  |  | Столяров Н.С. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

**1.Цель работы:** Изучить вычисление математических функций в с++.  
**2.Задачи работы:** 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.

  
**3.Описание функций:  
1)Имя: main**   
Назначение: Запрос координат точки от пользователя и определение находится ли она в области допустимых значений.  
Входные данные: нет.   
Выходные данные: нет.  
Побочный эффект: отсутствует.   
Тестовые данные:

|  |  |  |
| --- | --- | --- |
| X | Y | Ответ |
| 1 | 1 | Точка находится в области |
| 0.2 | -0.4 | Точка находится в области |
| 10 | -12 | Точка находится вне области |

Прототип: int main()  
Псевдокод:  
Функция проверки ввода данных read\_double()

Функция проверки ввода данных read\_double()

IF точка входит в фигуру THEN

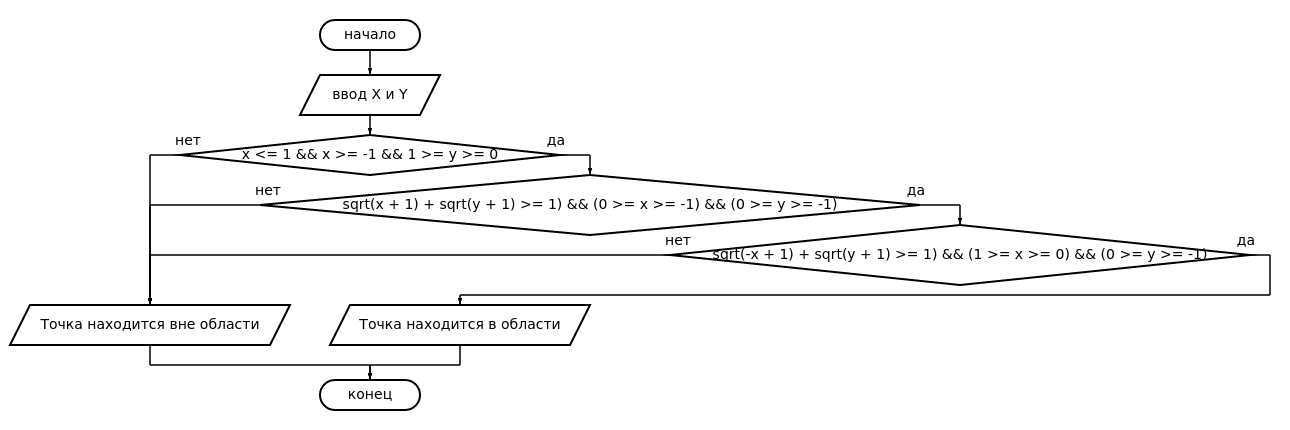
вывод «Точка находится в области»

ELSE

вывод «Точка находится вне области»

ENDIF

Вывод в консоль значения z\_2  
Сравнение z\_1 и z\_2  
Блок-схема:



**2)Имя: read\_double**   
Назначение: проверка переменной a на корректность ввода  
Входные данные: a(значение в градусах).   
Выходные данные: a(значение в градусах).  
Побочный эффект: отсутствует.   
Прототип: double read\_double()  
  
Псевдокод:  
Цикл продолжается до тех пор, пока пользователь не введет корректное значение

Ввод числа x

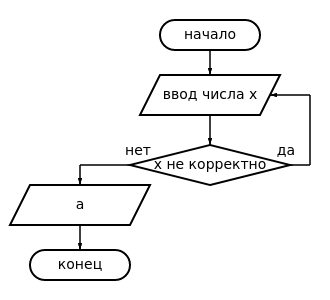
Если предыдущее извлечение оказалось неудачным, то...

Возвращаем scanf в 'обычный' режим работы(обнуление битов состояния)

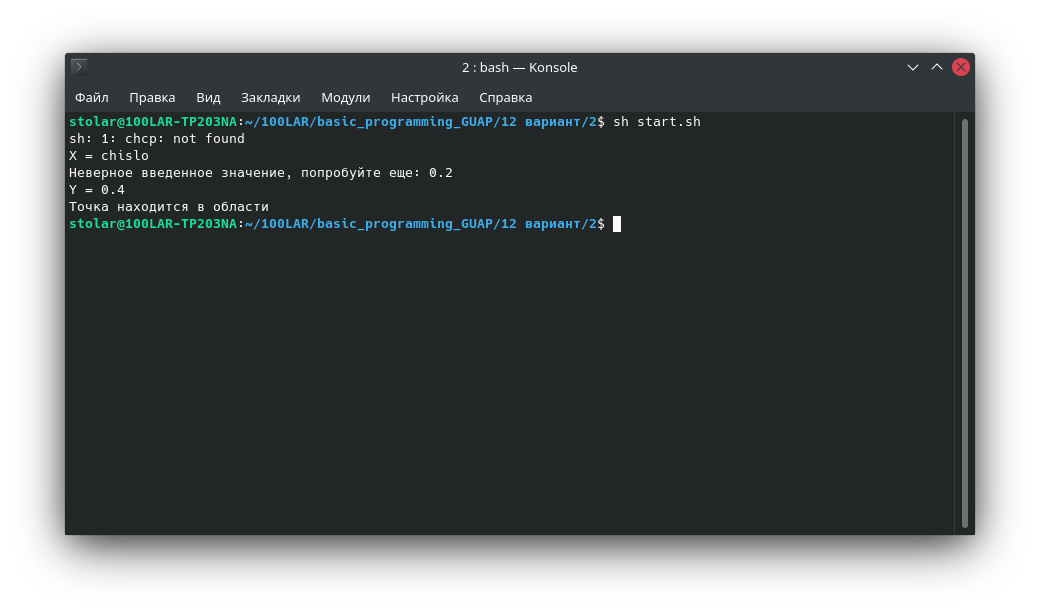
Удаляем значения предыдущего ввода из входного буфера

Если всё хорошо, то возвращаем x

Возврат значения x  
Блок-схема:

  
  
**9.ЛИСТИНГ КОДА:**/\*   
 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.   
\*/   
  
  
#include <stdio.h>   
#include <stdlib.h>   
#include <locale.h> // Для смены языка   
#include <math.h>   // Математические функции и константы   
  
// Функция для проверки ввода   
double read\_double(){   
       double x;   
 while ( (scanf("%lf",&x) ) != 1 ) {   
     printf("Неверное введенное значение, попробуйте еще: ");   
     while(getchar() != '\n');   
 }   
 return x;   
}   
  
// основной код   
int main() {   
       // смена кодировки   
 system("chcp 65001");   
  
 // ввод X   
 printf("X = ");   
 double x = read\_double();   
  
 // ввод Y   
 printf("Y = ");   
 double y = read\_double();   
  
 // Огромное условие   
 if (   
     (x <= 1 && x >= -1 && 1 >= y >= 0) &&   
     (sqrt(x + 1) + sqrt(y + 1) >= 1) && (0 >= x >= -1) && (0 >= y >= -1) &&   
     (sqrt(-x + 1) + sqrt(y + 1) >= 1) && (1 >= x >= 0) && (0 >= y >= -1)   
   ) {   
   printf("Точка находится в области\n");   
  
 } else {   
   printf("Точка находится вне области\n");   
  
 }   
  
 return 0;   
}

**10.Пример выполнения программы:**



Видно, что результаты расчётов совпадают с тестовыми данными.

**11.Анализ результатов и выводы:**

В ходе этой лабораторной работы мы научились работать с логическими операциями (в данном случае вспомнили)