|  |
| --- |
| Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования |
| Московский Государственный Технический Университет имени Н.Э. Баумана |
|  |

Факультет              ФУНДАМЕНТАЛЬНЫЕ НАУКИ

Кафедра             МАТЕМАТИКА И МАТЕМАТИЧЕСКАЯ ФИЗИКА

**Отчет по лабораторной работе № 2 и № 3**

**по курсу «Информатика»**

Студента           Борисов Иван Дмитриевич ­­­­­­­­­­­

(фамилия, имя, отчество)

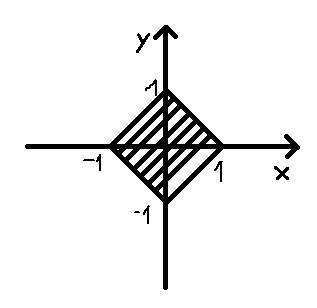
Группа                                          ФН11-12Б

Преподаватель \_   доцент, к.т.н. Ничушкина Т. Н.

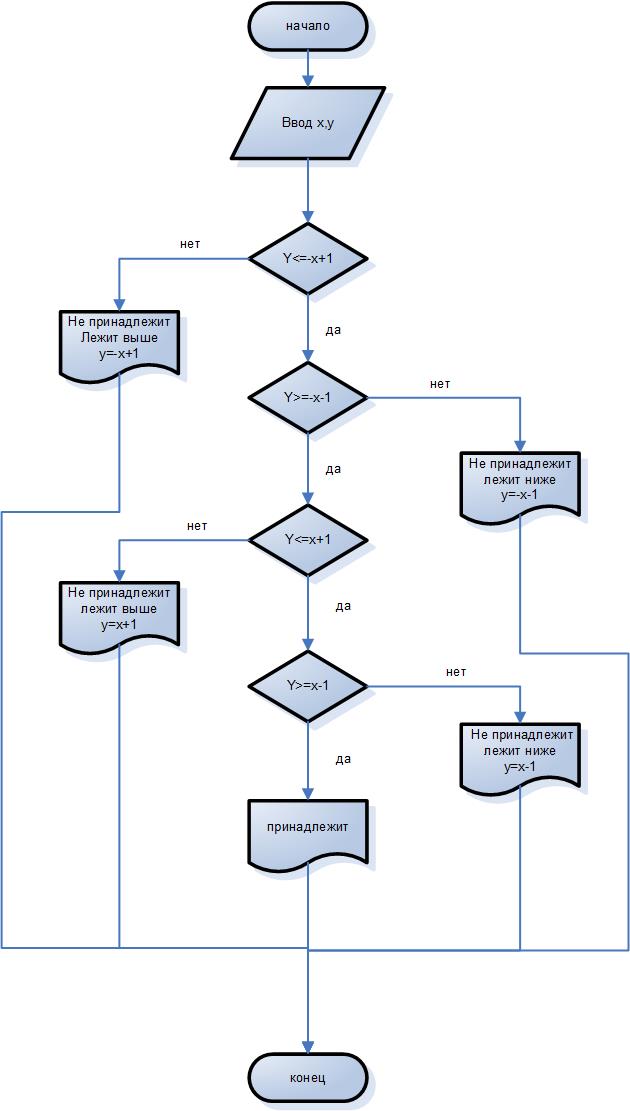
Должность, ФИО, подпись

2018

**ЛР 2. Программирование разветвляющегося вычислительного процесса.**

Даны вещественные числа X и Y. Определить принадлежит ли точка с координатами (X,Y) заштрихованной части плоскости. Протестировать все ветви алгоритма. 

**Схема алгоритма**

****

**Текст программы**

#include "stdafx.h"

#include <stdio.h>

#include <conio.h>

#include <locale.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

setlocale (0, "russian");

float x,y;

puts ("введите x,y");

scanf ("%f %f", &x,&y);

printf ("x=%5.2f y=%5.2f \n" ,x,y);

if (y<=-x+1)

if (y>=-x-1)

if (y<=x+1)

if (y>=x-1)

puts ("точка принадлежит");

else puts ("не принадлежит,лежит ниже y=x-1");

else puts ("не принадлежит,лежит выше y=x+1");

else puts ("не принадлежит,лежит ниже y=-x-1");

else puts ("не принадлежит,лежит выше y=-x+1");

puts ("нажмите любую клавишу для завершения");

getch();

return 0;

}

**Таблица тестовых примеров**

|  |  |  |  |
| --- | --- | --- | --- |
| **X** | **Y** | **ОЖИДАЕМЫЙ РЕЗУЛЬТАТ** | **ВЫВОД ПРОГРАММЫ** |
| **0** | **0** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **0** | **1** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **0** | **-1** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **1** | **0** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **1** | **1** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит выше y=-x+1** |
| **1** | **-1** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит ниже y=x-1** |
| **0** | **-2** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит ниже y=-x-1** |
| **0** | **2** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит выше y=-x+1** |
| **-1** | **0** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **-1** | **-1** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит ниже y=-x-1** |
| **-1** | **1** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит выше y=x+1** |
| **-2** | **0** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит ниже y=-x-1** |
| **2** | **0** | **НЕ ПРИНАДЛЕЖИТ** | **не принадлежит,лежит выше y=-x+1** |
| **0.5** | **0.5** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **0.5** | **-0.5** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |
| **-0.5** | **0.5** | **ПРИНАДЛЕЖИТ** | **ПРЕНАДЛЕЖИТ** |
| **-0.5** | **-0.5** | **ПРИНАДЛЕЖИТ** | **ПРИНАДЛЕЖИТ** |

**Вывод**

Я научился программировать разветвляющийся вычислительный процесс.

**ЛР 3. Программирование циклического процесса. Типы циклов.**

Рассмотреть решение предложенной задачи с использованием всех трех видов

циклов. Реализовать и отладить программу с наиболее рациональным вариантом цикла.

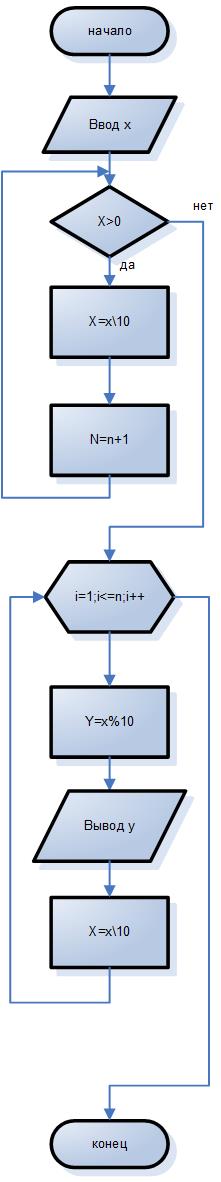
Обосновать выбор.

Определить число, получаемое записью в обратном порядке цифр заданного

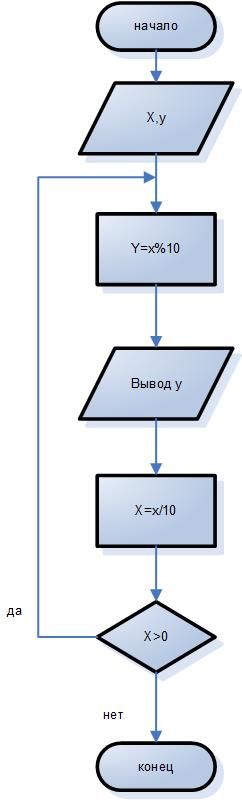
натурального числа. Массивы и строки не использовать.

**Схема алгоритма**

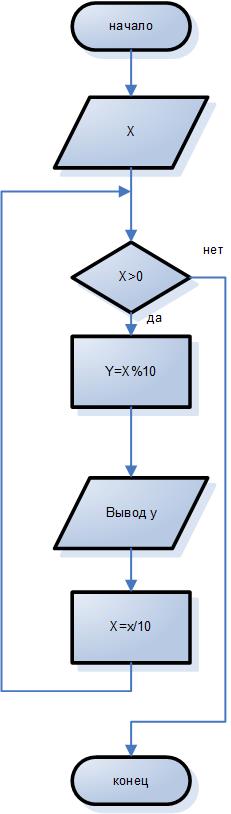
**Цикл FOR**

****

**Цикл DO WHILE**

****

**Цикл WHILE**

****

**Текст программы с помощью цикла WHILE**

#include "stdafx.h"

#include <stdio.h>

#include <conio.h>

#include <locale.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

setlocale (0, "russian");

long x,y;

puts ("введите x");

scanf ("%d", &x );

printf ("x=%d " ,x);

printf ("y=");

while (x>0)

{

y=x%10; printf("%d",y);

x=x/10;

}

puts ("\n нажмите любую клавишу для завершения");

getch();

return 0;

}

**Таблица тестовых примеров**

|  |  |  |
| --- | --- | --- |
| **Х** | **ОЖИДАЕМЫЙ РЕЗУЛЬТАТ** | **ВЫВОД ПРОГРАММЫ** |
| **123456789** | **987654321** | **987654321** |
| **987654321** | **123456789** | **123456789** |
| **543219876** | **678912345** | **678912345** |
| **678912345** | **543219876** | **543219876** |
| **135792468** | **864297531** | **864297531** |
| **864297531** | **135792468** | **135792468** |

**Вывод**

Я научился программировать циклический процесс тремя типами циклов.

Я реализовал и отладил программу с использованием цикла while, потому что посчитал его наиболее рациональным. Свой выбор я могу обосновать так: в цикле for сначала нужно найти количество цифр в вводимом числе и для написания программы у меня ушло бы больше времени, а в цикле do while мне не нравится, что сначала компьютер «прогоняет» один раз все действия с числом, а только потом смотрит на условие, и мне удобнее сначала ввести условие, а уже потом задавать действия при вводимом условии, что и происходит при использовании цикла while (я считаю это более наглядным и упорядоченным).