МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Физико-технический факультет

Кафедра вычислительной техники и электроники (ВТиЭ)

Лабораторная работа № 03

**Программирование с использованием циклов**

Выполнил студент 585 гр.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Г.Е. Хлыстов

Проверил: к.т.н,, доцент каф. ВТиЭ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ю.Г. Скурыдин

Лабораторная работа защищена

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2018 г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Барнаул 2018

**Задача №1**

1. **Формулировка задачи**

Дано 15 вещественных чисел. Определить, образуют ли эти числа возрастающую последовательность.

1. **Постановка задачи**

На вход поступает последовательность из 15 вещественных чисел. Программа выполняет проверку введенных чисел на упорядоченность по возрастанию. На выход поступает текстовое сообщение, в которой будет говориться является ли последовательность возрастающей или нет.

1. **Математическая модель**

В возрастающей последовательности последующее число больше предыдущего, из этого выходит, что числа последовательности нужно сравнивать с предыдущими.

1. **Описание алгоритма**

**Начало алгоритма**

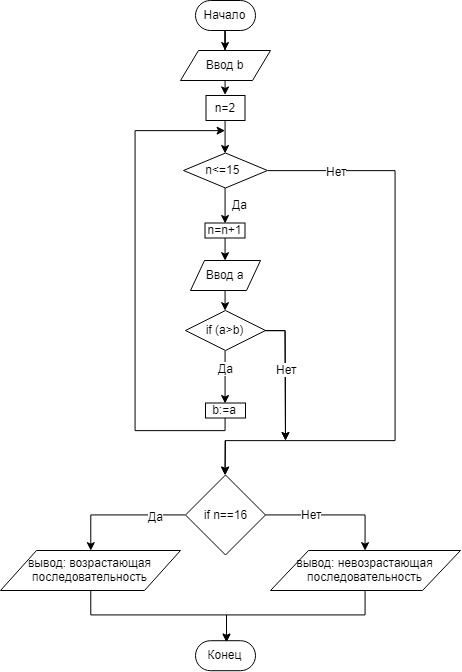
1. На вход поступает b
2. Числу n присваивается значение 1
3. Проверка условия: n<=15, если условие не подтверждается, то переходит на 8 пункт
4. Числу n присваивается значение n+1
5. На вход поступает число a
6. Проверка условия: a>b, если условие не подтверждается, то цикл прерывается и переходит на 8 пункт
7. Числу b присваивается значение a, переходим на 3 пункт
8. Проверка условия: n==16, если условие подтверждается то выводится " возрастающая последовательность ", а если нет то "невозрастающая последовательность"
9. Вывод текстового сообщения

**Конец алгоритма**

1. **Проект программы с определением замкнутых программных единиц и необходимых структур данных**

В данном случае программа состоит из четырех частей – 2 блоков ввода, блок сравнения и блока вывода.

1. **Опорный граф (блок-схема) алгоритма**



1. **Текст программы:**

**на языке С++**

#include <iostream>

#include <conio.h>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

setlocale(0, "");

int a, b, n;

cout << "введите число ";

cin >> b;

for (n=2; n<=15; n++ )

{

cout <<"введите число ";

cin >> a;

if (a>b)

b=a;

else break;

}

if (n==16)

cout <<" возрастающая последовательность "<< endl;

else cout << "невозрастающая последовательность" << endl;

\_getch();

return 0;

}

**на языке Pascal**

program project1;

var a, b:real;

n:integer;

begin

write ('введите число ');

read (b);

for n:=2 to 15 do

begin

write ('введите число ');

read (a);

if a>b then b:=a

else break

end;

if n=15 then write ('возрастающая последовательность')

else write ('невозрастающая последовательность');

read (a);

end.

1. **Проверка работоспособности (тестирование) программы**

Для проверки работоспособности используем пример, при использовании последовательности (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15). Программа выдает 'возрастающая последовательность', что является верным выводом программы.

1. **Сравнительный анализ и оценка эффективности работы программ на разных языках программирования**

Ввиду простоты программы она одинаково эффективно выполняется для обоих языков программирования.

**Задача №2**

1. **Формулировка задачи**

Написать программу для вычисления количества точек с целочисленными координатами, попадающих в круг радиуса r (r>0) с центром в точке с координатами x\_c и y\_c, заданными с клавиатуры.

1. **Постановка задачи**

На вход поступает 3 вещественных числа, 2 из которых определяют координату центра круга, а 1 радиус этого круга, который должен быть больше 0. Программа считает количество точек входящих в круг. На выход идет положительное целое число, являющееся количеством точек в этом кругу.

1. **Математическая модель**

Для нахождения точек в кругу, нужно, начиная с координат x=1 и y=1, проверять наличие точек, последовательно изменяя их координаты на 1. Для проверки нахождения точки в круге используется неравенство (x\*x)+(y\*y)<=r\*r. Например: при радиусе равном 3, мы подставляем координаты x и y начиная от значения равной единице – 1\*1+1\*1<=3\*3. Далее мы изменяем значение координаты x до того момента пока неравенство будет верным (1,2,3…), позже изменяется значение координаты y по такому же правилу. Таким образом будут подсчитаны все точки в одной четверти круга. Далее нужно учесть это количество точек для всех четвертей и прибавить точки, находящиеся на осях координат.

1. **Описание алгоритма**

**Начало алгоритма**

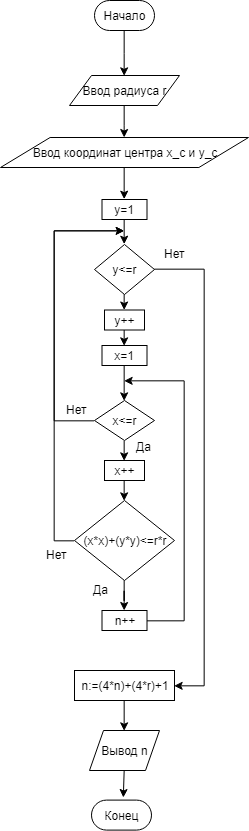
1. Ввод радиуса r
2. Ввод координат центра x\_с и y\_с
3. x=1
4. y=1
5. Проверка условия: y<=r, если не подтверждается то переходит к пункту 11
6. y++
7. Проверка условия: x<=r, если не подтверждается то переходит к пункту 5
8. x++
9. Проверка условия: (x\*x)+(y\*y)<=r\*r, если не подтверждается то переходит к пункту 5
10. n++, переходит к пункту, 7
11. n=4\*n+4\*r+1
12. Вывод количества точек n

**Конец алгоритма**

1. **Проект программы с определением замкнутых программных единиц и необходимых структур данных**

В данном случае программа состоит из четырех частей – 2 блоков ввода, 3 блока сравнения и блока вывода.

1. **Опорный граф (блок-схема) алгоритма**

****

1. **Текст программы:**

**на языке С++**

#include <iostream>

#include <conio.h>

using namespace std;

int main() {

setlocale(LC\_ALL, "Rus");

setlocale(0, "");

int x\_c, y\_c, r, x, y, n;

cout<< "введите радиус r " << endl;

cin >> r;

cout << "введите координаты центра x и y" << endl;

cin >> x\_c >> y\_c;

for (y=1; y<=r; y++)

for (x=1; x<=r; x++) {

if

n++;

}

n=4\*n+4\*r+1;

cout<<n; }

**на языке Pascal**

program project1;

var x, y, r, n:integer;

begin

n:=0;

write ('введите радиус r ');

read (r);

write ('введите координаты центра ');

read (x, y);

for x:=1 to r do

for y:=1 to r do

if ((x\*x)+(y\*y)<=r\*r) then

n:=n+1;

n:=(4\*n)+(4\*r)+1;

writeln ('всего точек в круге ', n);

readln (r);

end.

1. **Проверка работоспособности (тестирование) программы**

Для проверки работоспособности воспользуемся, примером, если радиус равен 5, а координаты (0,0), программа выдает число 81, что является верным исходом.

1. **Сравнительный анализ и оценка эффективности работы программ на разных языках программирования**

Ввиду простоты программы она одинаково эффективно выполняется для обоих языков программирования.