МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

РУТ(МИИТ)

Кафедра “УТБиИС”

**Отчёт №7**

**по дисциплине Информатика**

**Вариант 19**

Выполнил: Минаева Полина Валерьевна

Группа: УТН-111

Проверил: Новиков А.И.

МОСКВА 2020

**Задание**

1. Проинициализировать массив. Описать функцию, которая определит, образуют ли элементы массива арифметическую прогрессию. Если да, вернуть знаменатель прогрессии, иначе 0. Описать функцию, которая определит, образуют ли элементы массива геометрическую прогрессию. Если да, вернуть знаменатель прогрессии, иначе 1.
2. Описать функции ввода и вывода элементов массива. Описать функцию, которая удалит из массива первый и последний элементы.

**Код на языке C++**

**#include <iomanip>**

#include <iostream>

using namespace std;

void read\_mas(double\* mas, int size)

{

for (int i = 0; i < size; i++)

{

cout << "[" << i + 1 << "] = ";

cin >> mas[i];

}

}

void write\_mas(double\* mas, int size)

{

for (int i = 0; i < size; i++)

{

cout << setw(3) << setprecision(5) << mas[i] << " ";

}

cout << "\n";

}

double arifm(double\* mas, int size)

{

double d = mas[1] - mas[0];

for (int i = 2; i < size; i++)

{

if (mas[i] - mas[i - 1] != d)

{

return 0;

}

}

return d;

}

double geom(double\* mas, int size)

{

double q = mas[1] / mas[0];

for (int i = 2; i < size; i++)

{

if (mas[i] / mas[i - 1] != q)

{

return 1;

}

}

return q;

}

void remove\_start\_end(double\*\* mas, int& size)

{

double\* src = \*mas;

double\* m = new double[size - 2];

for (int i = 1; i < size - 1; i++)

{

m[i - 1] = src[i];

}

size = size - 2;

\*mas = m;

}

void demonstrate(double\* mas, int size)

{

cout << "Mas: ";

write\_mas(mas, size);

double d = arifm(mas, size);

if (d == 0)

{

cout << "Массив не образует арифметическую прогрессию.\n";

}

else

{

cout << "Массив образует арифметическую прогрессию. D = " << d << "\n";

}

double q = geom(mas, size);

if (q == 1)

{

cout << "Массив не образует геометрическую прогрессию.\n";

}

else

{

cout << "Массив образует геометрическую прогрессию. Q = " << q << "\n";

}

}

int main()

{

setlocale(LC\_ALL, "russian");

cout << "Выполнила Минаева Полина Валерьевна УТН - 111\n";

cout << "Программа для управления массивом с помощью функций\n";

int size = 10;

double\* mas;

mas = new double[size]{1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0};

demonstrate(mas, size);

mas = new double[size]{2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256.0, 512.0, 1024.0};

demonstrate(mas, size);

mas = new double[size]{7, 17, 54, 12, 91, 87, 39, 56, 90, 37};

demonstrate(mas, size);

remove\_start\_end(&mas, size);

cout << "После удаления первого и последнего элементов\nMas: ";

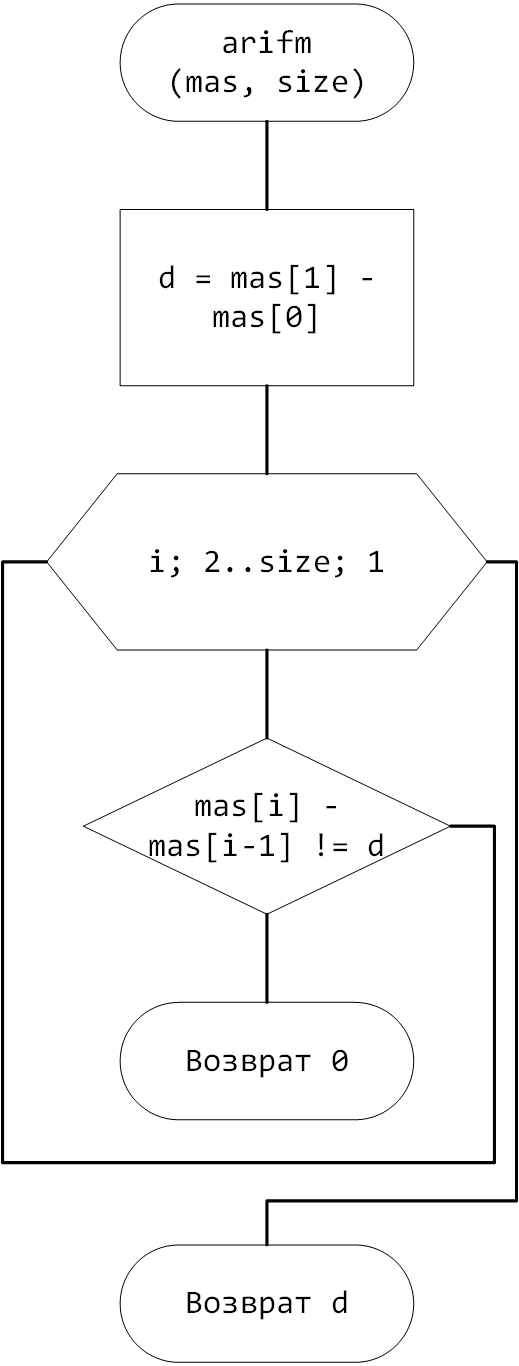
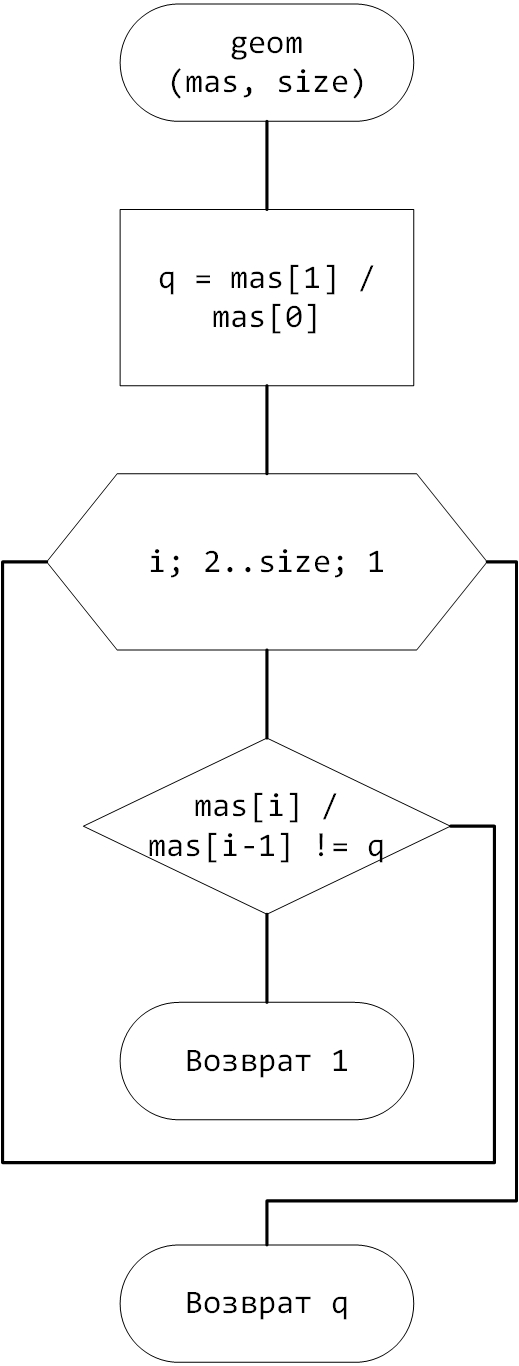
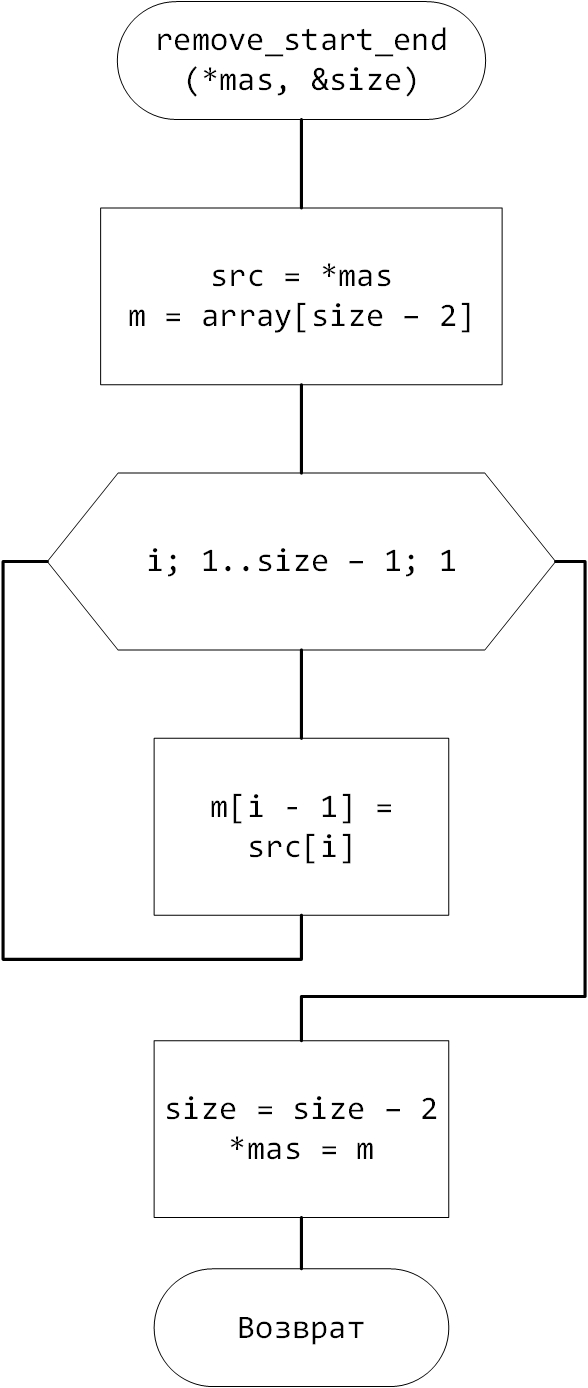
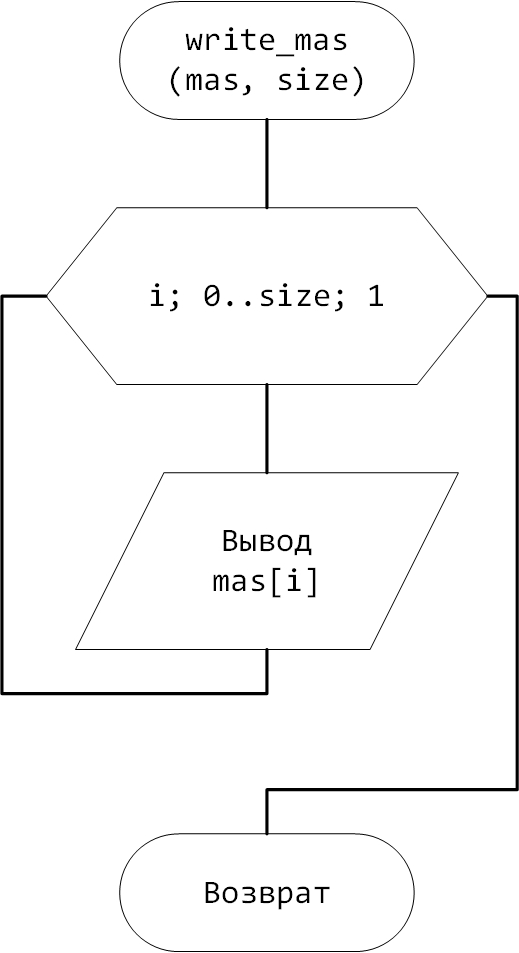
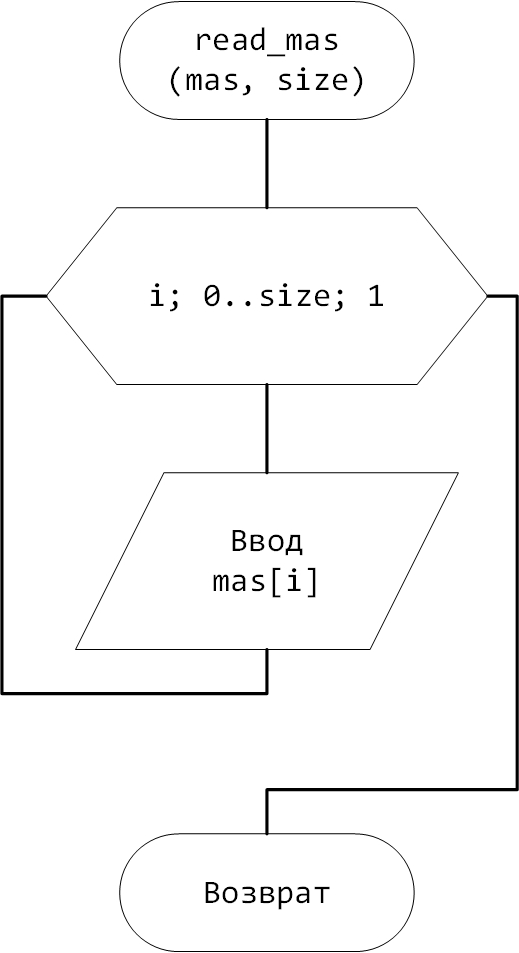
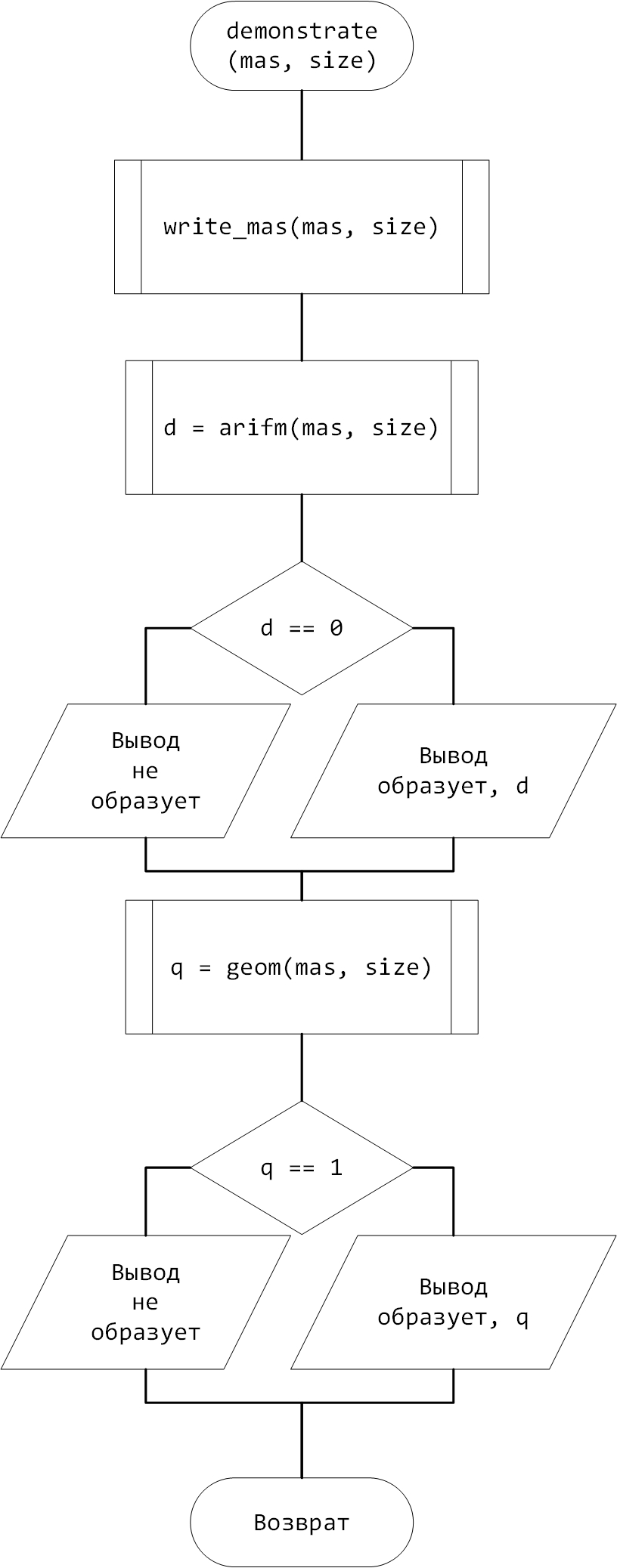
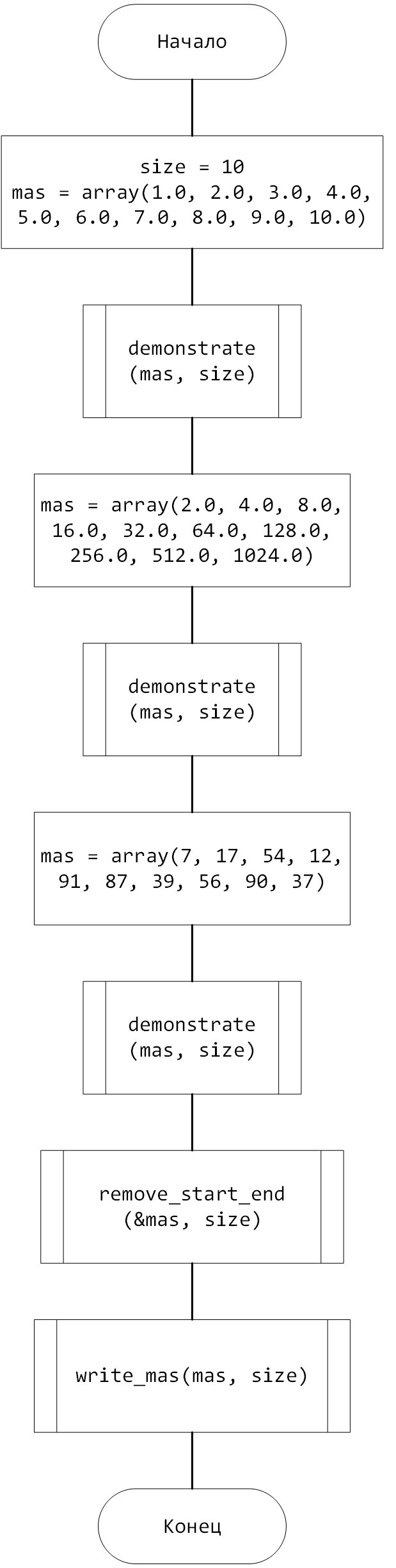
write\_mas(mas, size);

system("pause");

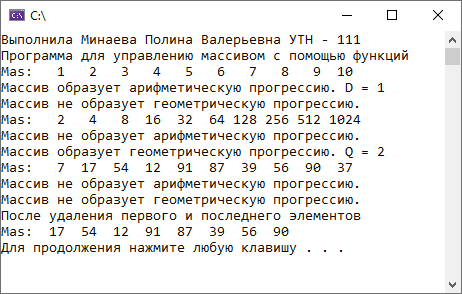
return 0;

}

**Блок-схема**



**Скриншоты работы программы:**



**Тестовые данные**

|  |  |
| --- | --- |
| Ввод | Вывод |
|  | Выполнила Минаева Полина Валерьевна УТН - 111  Программа для управления массивом с помощью функций  Mas: 1 2 3 4 5 6 7 8 9 10  Массив образует арифметическую прогрессию. D = 1  Массив не образует геометрическую прогрессию.  Mas: 2 4 8 16 32 64 128 256 512 1024  Массив не образует арифметическую прогрессию.  Массив образует геометрическую прогрессию. Q = 2  Mas: 7 17 54 12 91 87 39 56 90 37  Массив не образует арифметическую прогрессию.  Массив не образует геометрическую прогрессию.  После удаления первого и последнего элементов  Mas: 17 54 12 91 87 39 56 90 |

**Контрольный пример**

Изначальный массив {1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0}

1. При вызове arifm  
   d = 2.0 - 1.0 = 1, 3.0 – 2.0 = 4.0 – 3.0 = 5.0 – 4.0 = 6.0 – 5.0 = 7.0 – 6.0 = 8.0 – 7.0 = 9.0 – 8.0 = 10.0 – 9.0  
   Условие выполняется всегда, значит это арифметическая прогрессия со знаменателем = 1
2. При вызове geom  
   q = 2.0 / 1.0 = 2, 3.0 / 2.0 = 1.5 != 2  
   Условие не выполняется на первом шаге цикла, значит это не геометрическая прогрессия, возвращаем 1.

Изначальный массив { 2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256.0, 512.0, 1024.0}

1. При вызове arifm  
   d = 4.0 – 2.0 = 2, 8.0 – 4.0 = 4 != 2  
   Условие не выполняется на первом шаге цикла, значит это не арифметическая прогрессия, возвращаем 1.
2. При вызове geom  
   q = 4.0 / 2.0 = 2, 8.0 / 4.0 = 16.0 / 8.0 = 32.0 / 16.0 = 64.0 / 32.0 = 128.0 / 64.0 = 256.0 / 128.0 = 512.0 / 256.0 = 1024.0 / 512.0 = 2  
   Условие выполняется всегда, значит это геометрическая прогрессия со знаменателем = 2

Изначальный массив { 7, 17, 54, 12, 91, 87, 39, 56, 90, 37}

1. При вызове arifm  
   d = 17.0 – 7.0 = 10, 54.0 – 17.0 = 37 != 10  
   Условие не выполняется на первом шаге цикла, значит это не арифметическая прогрессия, возвращаем 1.
2. При вызове geom  
   q = 17.0 / 7.0 = 2.4285, 54.0 / 17.0 = 3.1764 != 2.4285  
   Условие не выполняется на первом шаге цикла, значит это не геометрическая прогрессия, возвращаем 1.

При вызове remove\_start\_end  
m = array[size – 2 = 10 – 2 = 8]  
m[0] = src[1], m[1] = src[2] … m[7] = src[8]  
Все значения поочерёдно копируются.

**Вывод**: На основании того, что полученные результаты совпали с ожидаемыми, программа работает верно.