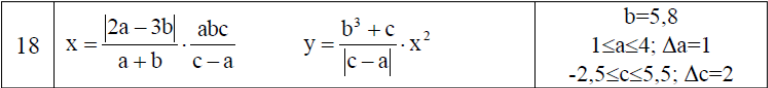
## Тема: «Разработка алгоритмов и программ со структурой вложенных циклов»

Вариант № 18

Задание: Разработать алгоритм для расчета заданных переменных.



Исходные данные: b, a1, a2, aa, c1, c2, cc.

Промежуточные данные: х.

Результат: значение y.

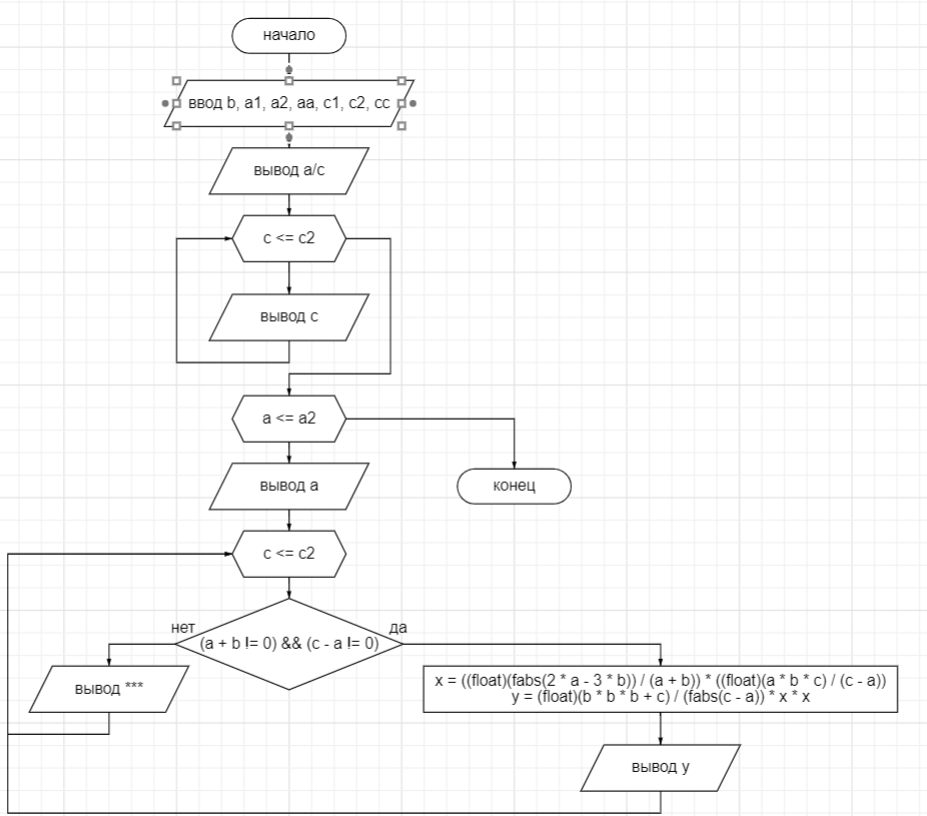
Ограничения:

1) Для вычисления переменной x по формуле соответствующей условию (2 \* a - 3 \* b)) / (a + b) проверять условие a + b != 0, иначе вывести «\*\*\*».

2) Для вычисления переменной x по формуле соответствующей условию

(a \* b \* c) / (c - a) проверять условие с - а != 0, иначе вывести «\*\*\*».

Схема алгоритма решения задачи имеет вид:



Текст программы имеет вид:

#include<iostream>

#include<iomanip>

#include<math.h>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RUSSIAN");

float y, x, b, a1, a2, aa, c1, c2, cc;

cout << "Введите b, a1, a2, aa, c1, c2, cc " << endl;

cin >> b >> a1 >> a2 >> aa >> c1 >> c2 >> cc;

cout << setw(6) << "a\\c";

for (float c = c1; c <= c2; c += cc)

{

cout << setw(20) << setprecision(2) << c;

}

cout << endl;

for (float a = a1; a <= a2; a += aa)

{

cout << setw(6) << a;

for (float c = c1; c <= c2; c += cc)

{

if ((a + b != 0) && (c - a != 0))

{

x = ((float)(fabs(2 \* a - 3 \* b)) / (a + b)) \* ((float)(a \* b \* c) / (c - a));

y = (float)(b \* b \* b + c) / (fabs(c - a)) \* x \* x;

cout << setw(20) << setprecision(10) << y;

}

else

{

cout << setw(10) << "\*\*\*" << endl;

}

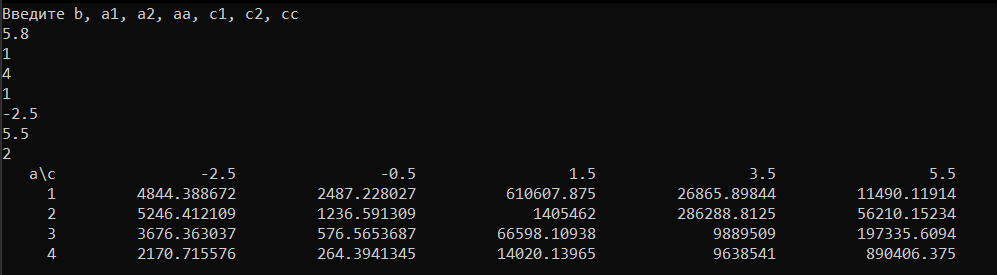
}

cout << endl;

}

}

Контрольное тестирование программы:



Вывод: Разработана учебная программа для решения варианта задания. Рассмотрена математическая модель решения, ограничения модели решения. Создан алгоритм программы.

1) **Алгоритм** — это последовательность команд, предназначенная исполнителю, в результате выполнения которой он должен решить поставленную задачу.

2)дискретностью, массовостью, формальностью, результативность, определенностью.

3) шестиугольник 

4) Циклическим называется алгоритм, содержащий циклы

5) Цикл – это алгоритмическая конструкция, которая представляет собой последовательность действий, повторяющихся многократно

6) Исполнение циклического алгоритма с безусловной структурой необходимо начать с задания переменной i исходного значения in. Далее следует выполнить проверку на превышение переменной i конечного значения iK. Если это превышение имеет место, то осуществляется завершение цикла и передача управления идущей за телом цикла команде.

7) Параметр цикла – это переменная, через которую организуется (описывается) цикл.

8) цикл с предусловием (цикл-пока), цикл с постусловием (цикл-до), цикл с параметром

9) for while

10) while

11) for

12) Составной оператор — конструкция языка программирования, состоящая из нескольких команд (операторов) языка программирования, но участвующая в программе в качестве единого оператора

If (условие)

{

//начало составного оператора

… //несколько операторов

//конец составного оператора

}

Else

{

//начало составного оператора

… //несколько операторов

//конец составного оператора

}

13) **Есть конструкция for:**

**for (init-expression; cond-expression; loop-expression)**

**действие**

**init-expression - выражение, которое выполняется лишь раз при старте цикла, например, чтобы задать начальное значение счетчика.**

**cond-expression - условие, которое проверяется перед каждой итерацией.**

**loop-expression - выражение, которое выполняется после каждой итерации.**

**Данный вид цикла может как выполняться, так и нет, в зависимости от истинности условия.**

**Есть конструкция while:**

**while (cond-expression)**

**действие**

**cond-expression - условие, которое проверяется перед каждой итерацией.**

**Данный вид цикла может как выполняться, так и нет, в зависимости от истинности условия.**

**Есть конструкция do-while:**

**do {**

**действие**

**} while (cond-expression)**

**cond-expression - условие, которое проверяется после каждой итерацией.**

**Данный вид цикла выполняется минимум один раз.**