Технический университет Молдовы.

Факультет информатики, вычислительной техники и микроэлектроники.

Кафедра прикладной информатики.

Лабораторные работы по

«Программирование на языке СИ».

Выполнил: ст. гр. IT-197. Макеев Никита.

Проверила: Раздорожная Ольга.

Кишинев 2019.

# Лабораторная работа №1.

## Тема: Использование операторов проверки условия и операторов цикла на языке Си.

## Цель работы:

Изучение приемов и способов использования операторов проверки условия и операторов цикла на языке Си для табулирования функции.

### Задание 1 Вариант 3:

Вычислить и вывести на экран в виде таблицы значения функции F на интервале от Хнач. до Хкон. с шагом dХ.

******

ax2 + bx + c при a < 0 и c ≠ 0

**F = ** при a > 0 и b = 0

a(x + c) в остальных случаях

где a,b,c - дствительные числа.

Функция F должна принимать действительное значение, если выражение АцИ(ВцИЛИСц) не равно нулю, и целое значение в противном случае. Через Ац, Вц и Сц обозна­чены целые части значений а, b, с, операции И и ИЛИ - поразрядные. Значения a, b, с, Хнач., Хкон., dХ ввести с клавиатуры.

### Выполнение задания:

#### Блок схема

Да

Вывод на экран результат

Условие 4

F3=

F2=

F1=

Нет

Да

Да

Нет

Условие 3

Условие 1

a, b, c , F, StartX, EndX, dX, k;

X <= EndX

Да

Нет

Условие 2

Да

Нет

Вывод на экран результат

X = x + dx

End.

#### Исходный код:

#include <math.h>

#include <stdio.h>

#include <conio.h>

int main(int argc, char\* argv[])

{

float a, b, c;

float StartX, EndX, dX;

float F;

int k;

printf("Enter:");

printf("\ta = "); scanf("%f", &a);

printf("\tb = "); scanf("%f", &b);

printf("\tc = "); scanf("%f", &c);

printf("\tX Start = "); scanf("%f", &StartX);

printf("\tX Final = "); scanf("%f", &EndX);

printf("\tdX = "); scanf("%f", &dX);

float x = StartX;

while (x <= EndX)

{

if ( a < 0 && c != 0 )

{

F = a \* x \* x + b \* x + c;

k = 1;

}

else if (a > 0 && b==0 && fabs(x-c) >.000001)

{

F = (-a) / (x - c);

k = 2;

}

else

{

F = a \* (x + c);

k = 3;

}

if (!(long)floor(a) || ((long)floor(b) && (long)floor(c)))

printf("x = %.2f\t F%d = %.0f\n", x, k,F);

else

printf("x = %.2f\t F%d = %.2f\n", x, k,F);

x += dX;

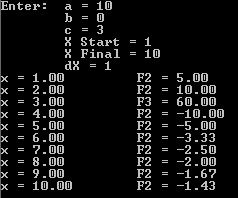
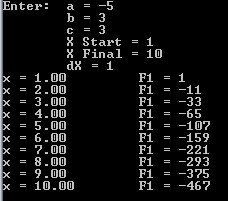
}

getch();

return 0;}

#### Результат:

По флагу 2 и 3: По флагу 1:

** **

### Задание 2 Вариант 3:

Вычислить и вывести на экран значения аргумента и значения функции, заданной несколькими выражениями, для заданного интервала хн ≤ x ≤ хк и шага dx изменения аргумента. Параметры a, b, c – действительные числа.

 при с < 0 и b ≠ 0

при с > 0 и b = 0

в остальных случаях

### Выполнение задания:

#### Исходный код программы

#include <math.h>

#include <stdio.h>

#include <conio.h>

int main(int argc, char\* argv[])

{

float a, b, c;

float StartX, EndX, dX;

float F;

int k;

printf("Enter:");

printf("\ta = ");

scanf("%f", &a);

printf("\tb = ");

scanf("%f", &b);

printf("\tc = ");

scanf("%f", &c);

printf("\tX Start = ");

scanf("%f", &StartX);

printf("\tX Final = ");

scanf("%f", &EndX);

printf("\tdX = ");

scanf("%f", &dX);

for (float x = StartX; x <= EndX; x += dX)

{

if ((c < 0) && (b != 0 ) && (fabs(x + b) > .000001 ))

{

F = (x+a) / (x + b);

k = 1;

}

else if (( c > 0 ) && ( b == 0 ))

{

F = (a \* x \* x) + (b \* b \* x) + c;

k = 2;

}

else if( c!=0 )

{

F = (sin(x) + a) / c;

k = 3;

}

else {

k = 4;

printf("x = %.2f\t F%d = Function is not declain\n", x, k);

continue;

}

if (!(long)floor(a) || ((long)floor(b) && (long)floor(c)))

printf("x = %.2f\t F%d = %.0f\n", x, k,F);

else

printf("x = %.2f\t F%d = %.2f\n", x, k,F);

}

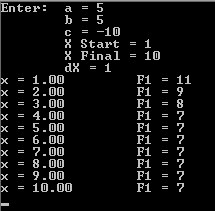
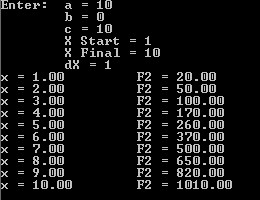
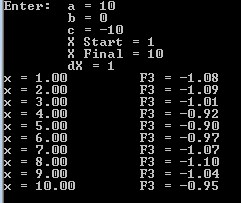
getch();

return 0;

}

#### Результат:

По флагу 1: По флагу 2: По флагу 3:

### Вывод:

В данной лабораторной работе были изучены:

1) Структура программы СИ

2) Операторы условного выполнения (if, else)

3) Операторы цикла (for , while)

4) Функции ввода вывода информации: printf, scanf

5)Табулирование функции f(x)

6)Работа с операциями побитного сравнения (&& и , || или, != не)

# Лабораторная работа 2

## Тема: Обработка одномерных массивов на языке Си

## Цель работы: Изучение возможностей и средств языка Си для программирования алгоритмов разветвленной и циклической структур при обработке одномерных массивов.

### Задание из Методички Вариант 1:

Для заданного одномерного массива из n элементов: Определить все максимальные элементы и их позиции в массиве, а также сумму отрицательных элементов.

### Задание из С Вариант 16:

В одномерном массиве, состоящем из *п* вещественных элементов, вычислить: 1) количество отрицательных элементов массива;2) сумму модулей элементов массива, расположенных после минимального по модулю элемента. Заменить все отрицательные элементы массива их квадратами и упорядочить элементы массива по возрастанию.

### Выполнение задания:

#### Исходный код:

int main(int argc, const char \* argv[]) {

int n = 5,i,array[n],tmp=0,neg = -1,sum=0,sum1 = 0;

printf("Заполнение массива \n");

for (i = 0; i<n; i++)

{

scanf("%d", &array[i]);

}

printf("Вывод массива на экран\n");

printf("a[n] = {");

for (i = 0; i<5; i++)

{

tmp = (array[i] < 0 ) ? tmp+1 : tmp+0;

printf("%d ", array[i]);

}

printf("}\n");

for (i = 0; i<5; i++)

{

sum1 = (array[i]<0) ? sum1+=array[i]: sum1+0;

}

printf("Сумма отрицательных элементов массива %d", sum1);

printf("\n");

printf("Количество отрицательных элементов массива: %d \n", tmp);

for (i=0; i < n; i++) {

if (array[i] < 0) {

neg = i;

break;

}

}

if (neg == -1) printf("нет отрицательных\n");

else {

printf("номер 1-го отриц: %d\n", neg+1);

sum = 0;

for (i=neg+1; i<n; i++) sum += abs(array[i]);

printf("сумма после него : %d\n", sum);

}

printf("Замена всех отрицательных элементов массива их квадратами \n");

printf("a[n] = {");

for (i = 0; i<5; i++)

{

array[i] = (array[i] < 0 ) ? array[i]\*array[i] : array[i];

printf("%d ", array[i]);

}

printf("}\n");

printf("Cортировка в поредяке возрастания\n");

printf("a[n] = {");

for (i = 0; i<5; i++)

{

for (int j = i+1; j < n; j++)

if (array[i] > array[j])

{

int buf = array[i];

array[i] = array[j];

array[j] = buf;

}

printf("%d ", array[i]);

}

printf("}\n");

printf("Находим в массиве максимальные элементы \n");

int imax1=0, imax2=0;

for (int i=0; i<n; i++) {

if (array[i]>=array[imax2]) {

imax1=imax2;

imax2=i;

} else if (array[i]>=array[imax1] && array[i]!=array[imax2])

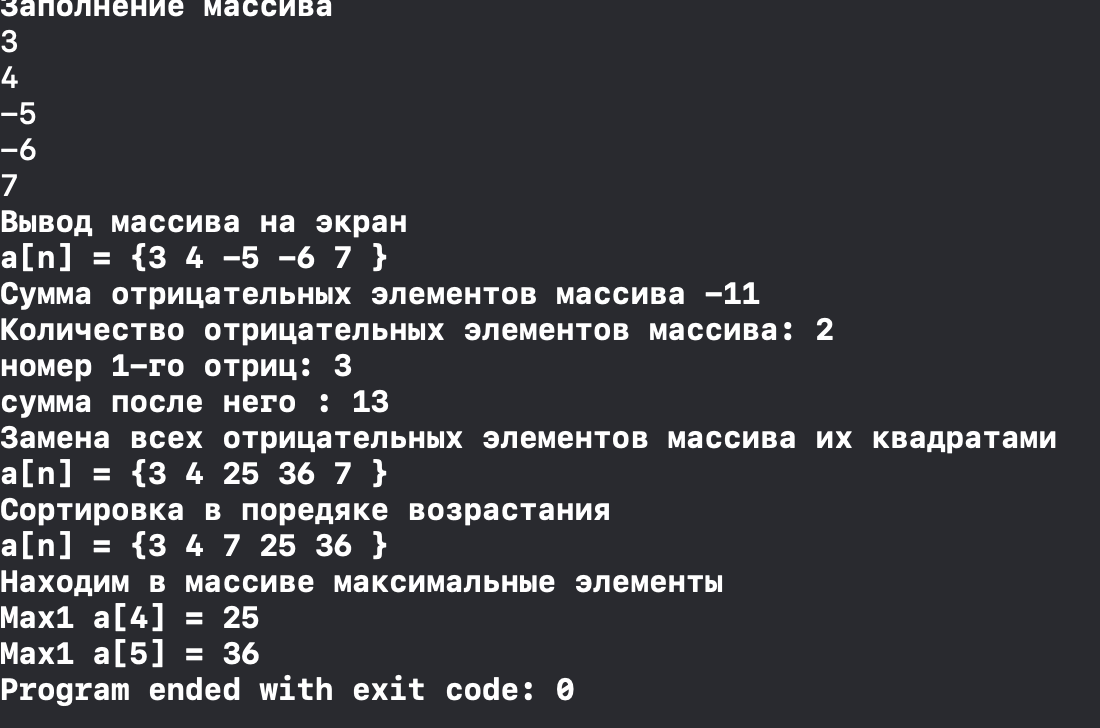
imax1=i;

}

printf("Max1 a[%d] = %d\nMax1 a[%d] = %d\n", imax1+1,array[imax1], imax2+1, array[imax2]);

return 0;

#### }Результат:

****

### Вывод:

В данной лабораторной работе были изучены:

1) Структура программы СИ

2) Тернарный оператор (?:)

3) Операторы цикла (for , while)

4) Функции ввода вывода одномерного массива: printf, scanf

Оглавление

[Лабораторная работа №1. 2](#_Toc23517301)

[Тема: Использование операторов проверки условия и операторов цикла на языке Си. 2](#_Toc23517302)

[Цель работы: 2](#_Toc23517303)

[Задание 1 Вариант 3: 2](#_Toc23517304)

[Выполнение задания: 2](#_Toc23517305)

[Задание 2 Вариант 3: 5](#_Toc23517306)

[Выполнение задания: 5](#_Toc23517307)

[Вывод: 6](#_Toc23517308)

[Лабораторная работа 2 7](#_Toc23517309)

[Тема: Обработка одномерных массивов на языке Си 7](#_Toc23517310)

[Цель работы: Изучение возможностей и средств языка Си для программирования алгоритмов разветвленной и циклической структур при обработке одномерных массивов. 7](#_Toc23517311)

[Задание из Методички Вариант 1: 7](#_Toc23517312)

[Задание из С Вариант 16: 7](#_Toc23517313)

[Выполнение задания: 7](#_Toc23517314)

[Вывод: 10](#_Toc23517315)