Лабораторная работа 3

Тема: Программирование циклических алгоритмов. Операторы цикла

Цель работы: изучить приемы составления циклических алгоритмов, правила использования операторов for, while, do – while.

ЗАДАНИЕ 1 (while)(task1): Вычислить и вывести на экран в виде таблицы значения функции F от -1 до 1 с шагом 0.05. Значения a, b и c вводятся с клавиатуры. Предусмотреть сообщение об ошибке в случае исключительной ситуации

Блок-схема

Начало

Готовим функции(оформление) под вывод таблицы

Ввод переменных x,F,a,b,c,step и вывод шапки таблицы

нет

если х<=1+1e-10

c>0 и x!=0 и a!=c)или(c<0 и x=0 и a!=c)или (c=0 и a!=c)

да

c < 0 и x != 0

c > 0 и x = 0

нет

нет

нет

да

да

да

выводим x и прочерки вместо F

F = b\*x/(c-a)

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

F = (-1)\*a \* x - c

Выводим значения x и F

x+=step

x+=step

Закрываем таблицу

Конец

Результат:

ЗАДАНИЕ 2 (for)(task2): Пользователь вводит число n (от 0 до 20). Сформировать соответствующую фигуру из n звездочек.

Блок-схема

Начало

Ввод переменной n

если 0<n<=20?

i<n?

выводим: n вне диапазона

нет

да

i++

i = j или i = (n - 1)-j или i = 0 или i = n – 1 или j=n/2?

нет

да

да

j<n?

нет

нет

да

выводим:” “(пробел)

выводим: \*

j++

Конец

Результат:

ЗАДАНИЕ 3 (схема)(task3): Для каждого x найти значения функции Y(x), суммы S(x) и |Y(x)-S(x)|. Результаты расчетов вывести в виде таблицы. Так как значение S(x) является рядом разложения функции Y(x), то значения S и Y для текущего аргумента x должны совпадать в целой части (в двух-четырех позициях после десятичной точки). Значения переменных установить сразу в коде: a = 0.1; b = 1.0; h = 0.1; n=20.

Блок-схема

Начало

Ввод переменных a,b,h,n

вывод шапки таблицы

нет

если х<=b+1e-10

да

вводим переменные s=0, i, f, p=x

если i<=2k+1

получаем f

нет

если k<=n?

нет

s+=p

да

p\*=x2k

да

вводим переменную f=1

f=f\*i

i++

p/=f

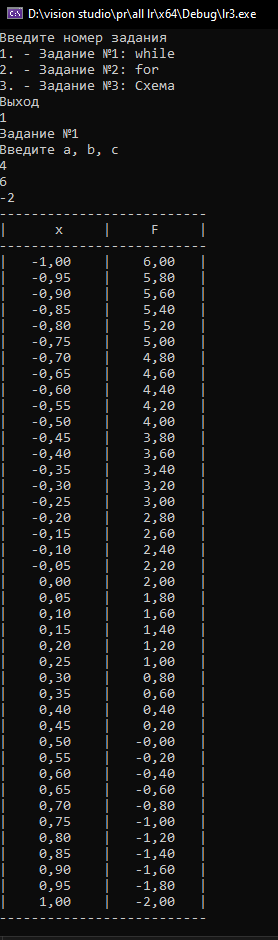
получаем y

Выводим значения x, s, y, |s-y|

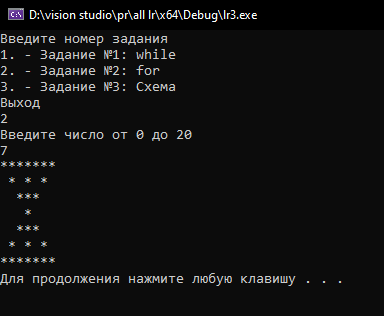
x+=h

Конец

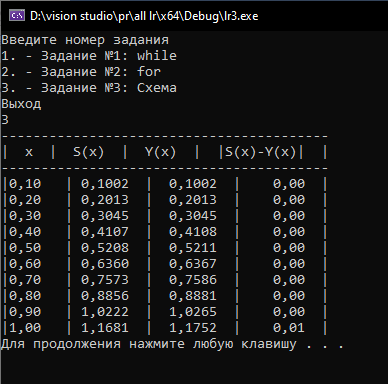
Результат task1:



Результат task2:



Результат task3:



#define \_USE\_MATH\_DEFINES

#include <iostream>

using namespace std;

void tableheader()

{

printf("--------------------------\n");

printf("| x | F |\n");

printf("--------------------------\n");

}

void tableline(double x, double F)

{

printf("|%8.2f |%8.2f |\n", x, F);

}

void tablefooter()

{

printf("--------------------------\n");

}

void task1()

{

cout << "Задание №1\n";

double x = -1, F, a, b, c, step = 0.05;

printf("Введите a, b, c\n");

scanf\_s("%lf%lf%lf", &a, &b, &c);

tableheader();

while (x <= 1 + 1e-10)

{

/\*if (x < 0 && b != 0)

{

F = a \* pow(x, 2) + b;

}

else if (x > 0 && b == 0 && (x - c) != 0)

{

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

}

else if (c != 0)

{

F = x / c;

}\*/

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

{

F = (-1)\*a \* x - c;

}

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

{

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

}

else if ((c>0 && x!=0 && a!=c)||(c<0 && x==0 && a!=c)|| (c==0 && a!=c))

{

F = b\*x/(c-a);

}

else

{

printf("|%8.2f | ----- |\n", x); x += step; continue;

}

tableline(x, F);

x += step;

}

tablefooter();

system("pause");

}

void task2()

{

int n;

cout << "Введите число от 0 до 20\n";

cin >> n;

int centr = n / 2;

if (0 < n <= 20)

{

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

{

for (int j = 0; j < n; j++)

{

//B-1 if (i == 0||i==n-1||j==0||j==n-1) cout << "\*";

//B-7 if (i == 0||i==n-1||j==0||j==n-1|| j==centr) cout << "\*";

//if (i == j || i == n - 1 - j);//B-6

if (i == j || i == (n - 1)-j || i == 0 || i == n - 1|| j==centr) cout << "\*";//B-2

else cout << " ";

}

cout << endl;

}

}

else cout << "n вне диапазона\n";

system("pause");

}

void task3()

{

double a = 0.1, b = 1, h = 0.1;

int n = 20;

printf("-----------------------------------------\n");

printf("| x | S(x) | Y(x) | |S(x)-Y(x)| |\n");

printf("-----------------------------------------\n");

/\*B-10

for (double x = a; x <= b + 1e-10; x += h)

{

double s = 0, p = 1;

for (int k = 1; k <= n; k++)

{

s += p;

p \*= pow(x, 2);

p /= 2 \* k \* (2 \* k - 1);

}

double y = (exp(x) + exp(-x)) / 2;

printf("|%3.2f |%7.4f |%8.4f |%8.2f |\n", x, s, y, fabs(y-s));\*/

for (double x = a; x <= b + 1e-10; x += h)

{

/\*В-13

double s = 0, p = -2 \* pow(x, 2);

int sign = -1;

for (int k = 2; k <= n; k++)

{

s += p;

p \*= sign \* 4 \* pow(x, 2);

p /= 2 \* k \* (2 \* k - 1);

}

double y = 2 \* (pow(cos(x), 2) - 1);\*/

//В-14

double s = 0, i, f, p = x;

for (int k = 1; k <= n; k++)

{

double f = 1;

for (i = 1; i <= 2\*k+1; i++)

{

f = f \* i;

}

s += p;

p \*= pow(x, 2 \*k);

p /= f;

}

double y = (exp(x) - exp(-x)) / 2;

printf("|%3.2f |%7.4f |%8.4f |%8.2f |\n", x, s, y, fabs(y - s));

}

system("pause");

}

void main()

{

setlocale(LC\_ALL, "russian");

do

{

system("cls");

int otvet;

cout << "Введите номер задания\n";

cout << "1. - Задание №1: while\n";

cout << "2. - Задание №2: for\n";

cout << "3. - Задание №3: Схема\n";

cout << "Выход\n";

cin >> otvet;

switch (otvet)

{

case 1: task1(); break;

case 2: task2(); break;

case 3: task3(); break;

case 0: exit(10);

default: cout << "Такого задания нет!\n";

}

} while (true);

}

Выводы: изучили приемы составления циклических алгоритмов, правила использования операторов for, while, do – while.