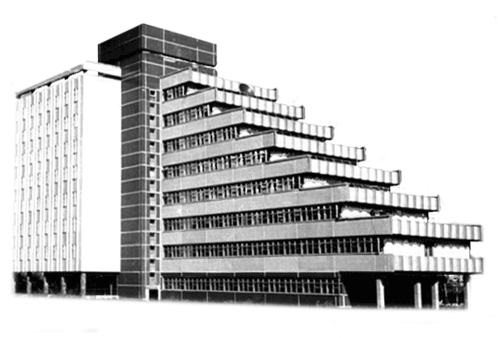
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное агентство по образованию

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Учебный Центр Информационных Технологий «Информатика»



Лабораторная работа № 3

по дисциплине «Информатика и программирование (I часть)»

Направление подготовки: 230105 - «Программное обеспечение вычислительной техники

и автоматизированных систем»

Выполнил слушатель: Коренский А.А.

Группа: 1

Вариант: 7

Дата сдачи: .03.2015

Преподаватель: Прытков Д.В.

Новосибирск, 2015г.

# Задание

Найти сумму степенного ряда , вычисляющего sh(x), для значений x = 0,1 – 1,0. Полученный результат сравнить с функцией sh(x) для таких же значений x.

# Теоретический материал

Итерационный цикл – цикл, в котором число его повторений и поведение программы на каждом шаге цикла зависят от результатов, полученных на предыдущих шагах Наиболее широко они применяются в вычислительной математике, когда для получения численного результата используется итерационный цикл последовательных приближений к нему.

Если попытаться изобразить общую схему итерационного цикла, то в нем обязательно будут переменные, сохраняющие результат предыдущего (x1) и еще более ранних (x2,...) шагов, а также переменная (x) -результат текущего шага:

// 1. Начальные значения предыдущих шагов

// 2. Условие завершения цикла

// 3. Следующий шаг

//---1------1-------2---------3-----3

for (x1=...,x2=...; l(x1,x2); x2=x1,x1=x)

{ ...x = f(x1,x2);...}

Если в итерационном цикле гарантируется выполнение одного шага, тo может быть использован цикл do...while:

x=...; x1=...; // Начальное значение текущего шага

do

{

x2 = x1; x1 = x; // Следующий шаг

x = f(x1,x2); // Результат текущего шага

} while (l(x2,x1,x)); // Условие завершения

Рассмотрим, как эта схема ложится на конкретные алгоритмы, использующие итерационные циклы.

Вычисление суммы степенного ряда

При вычислении сумм рядов, слагаемые которых включают в себя степени аргумента и факториалы, может быть также использован итерационный цикл. В этом цикле значение очередного слагаемого ряда вычисляется умножением предыдущего слагаемого на некоторый коэффициент, что позволяет избавиться от повторного вычисления степеней и факториалов. Сам коэффициент вычисляется таким образом:

Y = S0(x) + S1(x) + S2(x) +...+ Sn-1(x) + Sn(x) + ...

K (x,n) = Sn / Sn-1

Так, например, для ряда, вычисляющего sin(x), коэффициент и функция его вычисления имеют вид:

sin(x)=x – x3/3! + x5/5! – x7/7! +...+ (-1)n x2n+1 / (2n+1)!

S0 S1 S2 S3 ... Sn

k(x,n) = Sn / Sn-1 = - x2 / (2n (2n+1))

# Описание алгоритма

Цель данной программы – нахождение суммы степенного ряда для заданных значений x, и последующее визуальное сравнение найденных значений с функцией *sh(x)*. Алгоритм нахождения искомого значения представлен ниже.

На вход *h1()* получает значение формальной переменной *e*, необходимой для задания точности искомого значения. Функция *h1()* для достижения поставленной цели использует два цикла *for*. Первый цикл задает изменения по *x*, второй – реализует нахождение суммы степенного ряда при этих значениях.

# 4. Описание реализации

Подключаемые библиотеки:

#include "stdio.h"

#include "conio.h"

#include "stdlib.h"

#include "math.h"

Используемые функции:

int main() – вызывает функцию *h1()*;

printf() – вывод форматированного текста на экран;

system("cls") – для очистки экрана;

getch() – ожидание ввода символа с клавитуры.

Используемые конструкции:

for – оператор цикла;

# 5. Пример работы программы

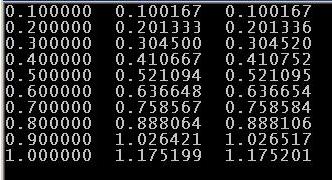


Рисунок 1. Результат работы функции *h1()*

На рисунке 1 показан результат работы функции *h1()*. Первый столбец слева представляет значения *x*, второй – значения суммы степенного ряда, третий – значения функции *sh(x)*.

**Выводы**

Перед выполнением этой работы, был изучен теоретический материал по данным темам. Затем необходимо было продумать алгоритм будущей программы, для общего представления ее конструкции. После чего, я приступил к написанию исходного кода.

В ходе выполнения лабораторной работы были изучены принципы итерационных процессов, также закреплены навыки работы с циклическими конструкциями.

После было проведено контрольное тестирование программы, которое показало правильность ее работы.

**Приложение. Текст программы с комментариями**

#include "stdio.h"

#include "conio.h"

#include "stdlib.h"

#include "math.h"

//Лабораторная работа №3 Вариант 7

//Задание: x+((x^3)/3!)+((x^5)/5!)+...((x^2n+1)/2n+1!)

double h1(float e);

int main() {

float e=0.0001;

system("cls");

h1(e);

getch();

}

double h1(float e) {

int n=1;

float x, S=0, Sn;

for(x=0.1;x<1.1;x+=0.1) {

for(Sn=x;Sn>e;n++) {

S+=Sn;

Sn\*=x\*x/(2\*n\*(2\*n+1));

}

printf("%f %f %f\n", x, S, sinh(x));

n=1;

S=0;

}

}

Защита:

1)

for (s=0, sn = x, n=1; fabs(sn) > eps; n++)

{ s += sn;

sn= - sn \* x / n;

}

x - x^2/1 + x^3/2 – x^4/3 + x^5/4 - …

2)

for (s=0, sn = x, n=1; fabs(sn) > eps; n+=2)

{ s += sn;

sn= sn \* x / (n\*(n+1));

}

x + x^2/2! + x^3/4! + x^4/6! + x^5/8! …