Министерство образования и науки Российской Федерации

Санкт-Петербургский государственный политехнический университет

—

Факультет технической кибернетики

**Кафедра «Информационная безопасность компьютерных систем»**

**ЛАБОРАТОРНАЯ РАБОТА №2**

**Вычисление синуса по формуле**

по дисциплине «Информатика»

Выполнил

студент гр.13508/13 А.Э.Палёный

<*подпись*>

Руководитель

доцент, к.т.н. А.И.Печёнкин

<*подпись*>

Санкт-Петербург

2015

Оглавление

[Цель работы: 3](#_Toc437295668)

[Решаемые задачи: 3](#_Toc437295669)

[Ход работы: 3](#_Toc437295670)

[Трудности, возникшие при разработке: 4](#_Toc437295671)

[Результат: 4](#_Toc437295672)

[Приложение: 4](#_Toc437295673)

# Цель работы:

Понять, как работает и как записывается в памяти тип данных double. Узнать из-за чего может произойти переполнение в программе.

# Решаемые задачи:

Вычисление значения функции синус по формуле

# Ход работы:

Мною была создана программа, состоящая из двух функций:

1. Вспомогательная функция, вычисляющая факториал, используя цикл:

for (r = 1; n > 1; r \*= (n--));

1. Основная функция, в которой происходят все вычисления. Она условно разбита на части:
   1. Тут происходит объявление всех основных переменных, их обнуление, а также получение данных от пользователя
   2. Далее расположен цикл for, в котором реализован подсчёт факториала и степенной функции, вывод их значений на экран, а также их деление и умножение на , и подсчёт суммы.
   3. В третьей части программы, происходит вывод получившихся результатов.

Результат работы программы для числа 5 и точности 0.001:

Здесь:  
Answer: -результат с точность введённой пользователем-

Answer:-результат вычисления по формуле до переполнения-

Sin = результат который должен получиться

# Трудности, возникшие при разработке:

Когда программа вычисляла значение части формулы, а именно факториала, то получала 15-значное число, а дальше нули. Из-за этого появлялась погрешность в вычислениях.   
При достаточно больших числах, введённых пользователем, это погрешность увеличивалась в разы, и программа выдавала неверный результат вычисления.

Почему же это происходит?  
Ответ на этот вопрос скрывался в описании типа double(основной тип данных, который использовался в программе). Оказывается, тип данных записывает числа в память иначе, нежели Long, Int, Short. Рассмотрим запись типа double.   
Во-первых, этот тип занимает в памяти 8 байт, что равно 64 битам, по этому:

| 1 бит под знак | 11 бит экспоненты | 52 бита мантиссы |

Присвоим переменным значение из каждой области для наглядности

| a | b | c |

Теперь покажем, как вычисляется значение типа double

Из этого видно, что вероятней всего переполнится переменная c. А под неё выделено всего лишь 52 бита, т.е. максимальное число, которое в неё поместится - . Т.е. точность, с которой считает программа это – 15 знаков. Но при любых числах, введённых пользователем, факториал рано или поздно достигнет гигантских масштабов, и его запись не будет влезать в 15 знаков хотя бы до точки. Тут-то и начнётся переполнение факториала, в результате которого погрешность будет огромная.

Так, что данная программа правильно считает при достаточно маленьких значениях, введённых пользователем.

# Результат:

Программа была создана в Visual Studio 2010 и успешно протестирована на моём компьютере. Но проблему, связанную с погрешностью исправить не удалось.

# Приложение:

#include <stdio.h>

#include <math.h>

#include <float.h>

double factorial (int n)

{

double r;

for (r = 1; n > 1; r \*= (n--))

;

return r;

}

int main()

{

double sum,bef,toch,levo,dop,sn;

int n;

float h,x;

//Delete Later

double i=0; //факториал

double j=0; //степенная функция

h=0;

sum=0;

bef=0;

toch=1;

levo=1;

printf("Vvedite x: ");

scanf("%g",&x);

printf("Vvedite tochnost': ");

scanf("%g",&h);

for(n=0; 1/j != 0 && 1/i != 0 ;n++)

{

bef=sum;

i=factorial(2\*n+1);

j=pow(x,2\*n+1);

printf("Factorial = %.21Lf\n",i);

printf("Stepen = %.21Lf\n",j);

dop=pow(-1,n)\*j/i;

sum=sum+dop;

// printf("dop = %.100lf\n",dop);

// printf("sum = %.100lf\n",sum);

//-----------расчётточности введённой пользователем------------

if (fabs(sum-bef) < h && toch == 1)

{

toch=sum;

levo=bef;

}

//-----------------------------------------------

}

//==========Проверка на точность============

if(toch==1 && levo==1)

printf("Izza perepolneniya s takoy tochnost'u\n programma ne mozhet schitat'");

else

{

printf("levo: %.100lf\n",levo);

printf("Answer: %.100lf\n",toch);

}

//==========================================

printf("\nAnswer: %.100G\n",bef);

sn=sin(x);

printf("sin = %.100lf\n",sn);

scanf("%g",&x);

return 0;

}