Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе №3**

Дисциплина: Основы алгоритмизации и программирования

Тема: “Динамические структуры данных”

Вариант 22

Выполнил:

Студент гр. ИВТ-20-2б

Чувашев Максим Алексеевич

Проверил:

Доцент кафедры ИТАС  
Викентьева О.Л.

Пермь, 2021

## **Цель работы**.

Практика в организации итерационных и арифметических циклов.

## **Постановка задачи**

Для х изменяющегося от a до b с шагом (b-a)/k, где (k=10), вычислить функцию f(x), используя ее разложение в степенной ряд в двух случаях:

а) для заданного n;

б) для заданной точности e (e=0.0001).

Для сравнения найти точное значение функции.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | функция | Диапазон  Изменения аргумента | n | сумма |
| 22 |  |  | 15 |  |

## **Анализ задачи**

1. Вспомогательные функции
   1. Определим функцию, которая будет возвращать значение функции от получаемого значения x.

static double F(double x)

{

return ((1 - (Math.Pow(x, 2) / 2)) \* Math.Cos(x) - (x / 2 \* Math.Sin(x)));

}

* 1. Определим функцию, которая будет возвращать значение факториала от получаемого аргумента.

static int Factorial(int n)

{

int s = 1;

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

{

s \*= i;

}

return s;

}

1. Главная функция
   1. Инициализируем необходимые переменные

Console.WriteLine("y = (1 - (x^2/2)) \* cos(x) - (x/2) \* sin(x))\n");

double a = 0.1, b = 1;

double k = (b - a) / 10;

int count = 0;

* 1. Внешний цикл, в котором создаем переменную x, приравниваем ее к a, x будет увеличиваться с шагом k, до тех пор, пока не станет больше либо равен b. Так же увеличиваем переменную count, которая будет служить для вывода количества различных x.

for (double x = a; x <= b; x += k, count++)

{

}

* 1. Два вложенных цикла, первый из которых будет искать сумму ряда при различных значениях n, пока n меньше заданного значения.

for (int n = 1; n <= 15; n++)

{

S += Math.Pow(-1, n) \* (2 \* Math.Pow(n, 2) + 1) / Factorial(2 \* n) \* Math.Pow(x, 2 \* n);

}

* 1. Второй из вложенных циклов ищет сумму элементов ряда, условием выхода из цикла является удовлетворение точности вычисления, если разность значений текущего с предыдущим значением будет равна или меньше заданной точности, то сумма ряда найдена.

for (int i = 1; Math.Abs(prevF - f) > ep; i++)

{

prevF = f;

f = Math.Pow(-1, i) \* (2 \* Math.Pow(i, 2) + 1) / Factorial(2 \* i) \* Math.Pow(x, 2 \* i);

SE += f;

}

## **Код программы**

using System;

namespace Lab\_3

{

class Program

{

public const double ep = 0.0001;

static double F(double x)

{

return ((1 - (Math.Pow(x, 2) / 2)) \* Math.Cos(x) - (x / 2 \* Math.Sin(x)));

}

static int Factorial(int n)

{

int s = 1;

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

{

s \*= i;

}

return s;

}

static void Main(string[] args)

{

Console.WriteLine("y = (1 - (x^2/2)) \* cos(x) - (x/2) \* sin(x))\n");

double a = 0.1, b = 1;

double k = (b - a) / 10;

int count = 0;

for (double x = a; x <= b; x += k, count++)

{

double S = 1.0;

for (int n = 1; n <= 15; n++)

{

S += Math.Pow(-1, n) \* (2 \* Math.Pow(n, 2) + 1) / Factorial(2 \* n) \* Math.Pow(x, 2 \* n);

}

double SE = 1, f = 0, prevF = 1;

for (int i = 1; Math.Abs(prevF - f) > ep; i++)

{

prevF = f;

f = Math.Pow(-1, i) \* (2 \* Math.Pow(i, 2) + 1) / Factorial(2 \* i) \* Math.Pow(x, 2 \* i);

SE += f;

}

Console.WriteLine($"[{count}] \tx = {x}; \tSN = {S}; \tSE = {SE}; \tY = {F(x)}\n");

}

}

}

}

## **Блок схема**







## **Работа программы**

