**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования   
"Национальный исследовательский университет   
"Высшая школа экономики"**

Московский институт электроники и математики Национального

исследовательского университета "Высшая школа экономики"

Департамент прикладной математики

**ОТЧЕТ**

**По лабораторной работе № 3**

**По курсу «Алгоритмизация и программирование»**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | |  | ФИО студента | | Номер группы | Дата |
| Индюченко Никита Андреевич | БПМ211 | 10.10.2021 |
|  |
|  |
|  |

**Москва – 2021 г.**

**ЗАДАНИЕ (вариант №12)**

**Вычислить приближенное значение функции, вычислив сумму конечного числа**

**элементов ряда двумя способами, используя разные типы циклов:**

**1) с заданной точностью;**

**2) для заданного количества членов ряда.**

**Переход к способу вычисления реализовать с помощью оператора выбора.**

*Текст задания Вашего варианта*

**

**РЕШЕНИЕ**

Код программы с комментариями

#include <stdio.h> //указывает и подключает расположенные файлы, библиотека потока ввода и вывода

#include <math.h> // библиотека математичских констант и функций

// протипы функций

long long int factorial(long long int \_n); // факториал

int choice(); // команда пользователя

void command\_1(); // команда 1, которая считает с заданной точностью

void command\_2(); // команда 2, которая счиает с заданным кол-вом членов последовательности

int main()

{

switch (choice())

{

case 1:

command\_1(); // выполение функции команда 1

break;

case 2:

command\_2(); // выполение функции команда 2

break;

default:

printf("The command does not exist\n"); // вывод в консоль, если пользователь ввёл параметр, которого не рассматривается выше

break;

}

}

int choice()

{

int x = 0; // целочисленная переменная, которая запоминает команду пользователя

printf("With the specified accuracy=1\nFor a given number of row members=2\nYour choice="); // вывод в консоль

scanf\_s("%d", &x); // считывание c клавиатуры x

return x; // возвращение x

}

void command\_1() // функция команда 1, которая считает выражение с заданной точностью

{

double tmp\_E, x, tmp\_sum=1; // объявление вещественных переменных эпсилон, числа x, которое будет использоваться в вычисление членов последовательности, а также инициализирование 1, из которой будут вычитаться дроби (члены)

printf("Enter E="); // ввод в консоль

scanf\_s("%lf", &tmp\_E); // считывание с клавиатуры эпсилон

printf("Enter x="); // ввод в консоль

scanf\_s("%lf", &x); // считывание с клавиатуры x

if (fabs(x) >= 1) // проверка на то, что x по модулю больше или равен 1

{

printf("Error x\n"); // данный x не подходит по условию задачи и выводится на консоль

}

else // иначе (если подходит по условию)

{

long long int i = 1; // целочисленная переменная, которая будет считать член последовательности, начиная с 1

while (1) // цикл, который будет бесконечный, пока в теле его не будет команды, которая выйдет из него (break)

{

double fraction = (factorial((2 \* i)) \* pow(x, i)) / ((1 - 2 \* i) \* pow(factorial(i), 2) \* pow(4, i)); // объявление вещественной переменной дробь, в которой считается i-ый член последовательности, используя математические функции, а также функция факториал

i++; // увеличивет i на 1

if (fabs(fraction) < fabs(tmp\_E)) // условие на то, что дробь по модулю меньше эпсилона

{

break; // тогда выходим из цикла while

}

else // иначе

{

tmp\_sum += fraction; // из переменной вычитаем полученную дробь

}

}

printf("%lf", tmp\_sum); // вывод полученной переменной

}

}

void command\_2() // функция команда 1, которая считает выражение с заданным кол-во членов последовательности

{

int tmp\_n; // целочисленная переменная, которая хранит в себе кол-во членов последовательности

printf("Enter n="); // ввод в консоль

scanf\_s("%d", &tmp\_n); // считывание с клавиатуры кол-во членов последовательности

double x, tmp\_sum = 1; // объявление вещественных переменных x и tmp\_sum

printf("Enter x="); // ввод в консоль

scanf\_s("%lf", &x); // считывание с клавиатуры параметра x

if (fabs(x) >= 1) // условие (проверка), что x по модулю больше или равен 1

{

printf("Error x\n");// вывод в консоль

}

else // иначе считает значение каждого члена последовательности

{

for (int i = 1; i <= tmp\_n; i++) // цикл, который считает n членов последовательности

{

double fraction = ((factorial((2 \* i)) \* pow(x, i))) / ((1 - 2 \* i) \* pow(factorial(i), 2) \* pow(4, i)); // объявление вещественной переменной дробь, в которой считается i-ый член последовательности, используя математические функции, а также функция факториал

tmp\_sum += fraction; // складываем полученную дробь с tmp\_sum

}

printf("%lf\n", tmp\_sum); // вывод полученной переменной

}

}

long long int factorial(long long int \_n) // функция факториал,

{

long long int fact\_sum = 1; // целочисленная переменная, которая будет хранить произвдение n чисел , начиная с 1

for (long long int j = 1; j <= \_n; j++) // цикл от 1 до n, включительно

{

fact\_sum \*= j;

}

return fact\_sum; // возвращение переменной

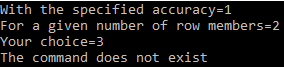
}

**ТЕСТЫ**

**Тест № 1**

**choice=3**

*Результаты теста 1*

**

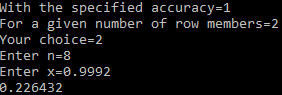
**Тест № 2**

**choice =2**

**n=8**

**x=0.9992**

*Результаты теста 2*

**

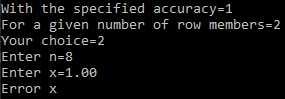
**Тест № 3**

**choice=2**

**n=8**

**x=1.00**

*Результаты теста 3*

**

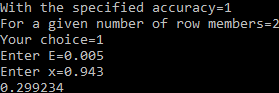
**Тест № 4**

**choice=1**

**E=0.005**

**X=0.943**

*Результаты теста 4*

**

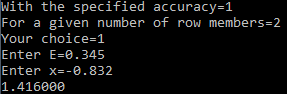
**Тест № 5**

**Choice=1**

**E=0.345**

**X=-0.832**

*Результаты теста 5*

****

**Тест № …**

*Результаты теста …*