МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский Авиационный Институт» (Национальный Исследовательский Университет)

Институт: №8 «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Курсовая работа

по курсу «Вычислительные системы» I семестр

Задание 3

«Вещественный тип. Приближенные вычисления. Табулирование функций»

|  |  |
| --- | --- |
| Группа | М8О-109Б-22 |
| Студент | Потапов Е.Д. |
| Преподаватель | Сысоев М.А. |
| Оценка |  |
| Дата |  |

Москва, 2022

# Постановка задачи

Составить программу на Си, которая печатает таблицу значений элементарной функции, вычисленной двумя способами: по формуле Тейлора и с помощью встроенных функций языка программирования. В качестве аргументов таблицы взять точки разбиения отрезка [a, b] на n равных частей (n+1 точка включая концы отрезка), находящихся в рекомендованной области хорошей точности формулы Тейлора.

Вычисления по формуле Тейлора проводить по экономной в сложностном смысле схеме с точностью ε \* 10k, где ε - машинное эпсилон аппаратно реализованного вещественного типа для данной ЭВМ, а k – экспериментально подбираемый коэффициент, обеспечивающий приемлемую сходимость. Число итераций должно ограничиваться сверху числом порядка 100. Программа должна сама определять машинное ε и обеспечивать корректные размеры генерируемой таблицы.

# Вариант 9:

Ряд Тэйлора:



Функция:



Значения а и b: 0.1 и 0.6

# Теоретическая часть

**Формула Тейлора** — формула разложения функции в бесконечную сумму степенных функций. Формула широко используется в приближённых вычислениях, так как позволяет приводить трансцендентных функций к более простым. Сама она является следствием теоремы Лагранжа о среднем значении дифференцируемой функции. В случае a=0 формула называется рядом Маклорена.



**Машинное эпсилон** — числовое значение, меньше которого невозможно задавать относительную точность для любого алгоритма, возвращающего вещественные числа. Абсолютное значение для машинного эпсилон зависит от разрядности сетки применяемой ЭВМ и от разрядности используемых при расчёте чисел. Формально это машинное эпсилон определяют как число, удовлетворяющее равенству 1 + ε = 1. Фактически, два отличных от нуля числа являются равными с точки зрения машинной арифметики, если их модуль разности меньше или не превосходит машинное эпсилон.

В языке Си машинные эпсилон определено для следующих типов: float – 1.19 \* 10-7, double – 2.20 \* 10-16, long double – 1.08 \* 10-19.

# Описание алгоритма

Рассмотрим алгоритм решения. Сперва нужно найти машинное эпсилон, на котором будет основываться точность вычисления. Это можно сделать просто деля 1 на 2.

Для каждой N+1 строки нужно просуммировать i членов формулы Тейлора, пока |A1-A2|> ε. Для этого просто ищем каждый новый член из формулы Тэйлора и суммируем с результатом

# Использованные в программе переменные

|  |  |  |
| --- | --- | --- |
| Название переменной | Тип переменной | Смысл переменной |
| n | Int | Количество разбиений отрезка |
| Iter | Int | Сколько рядов Тейлора |
| Ans | Double | Сумма ряда Тейлора |
| Func | Double | Значение функции |
| Add | Double | Член ряда Тейлора |
| L | Double | Левая граница отрезка |
| R | Double | Правая граница отрезка |
| x | Double | Просто икс |

**Исходный код программы:**

#include <stdio.h>

#include <limits.h>

#include <math.h>

long double factorial (long double n)

{

long double r;

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

return r;

}

int main()

{

int n, iter;

double ans, func, add, l = 0.1, r = 0.6, x = 0.1;

long double eps = 1.0l;

while (2.0l + eps / 2.0l > 2.0l) {

eps /= 2.0l;

}

printf("Machine eps double = %.16Le\n", eps);

printf("Write n: \n");

scanf("%d", &n);

printf("n = %d, \n", n);

printf("Table znacheniy Teylor and stand f(x) = e^2x");

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

printf("| x | sum | f(x) |count iter |\n");

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

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

add = 1;

iter = 1;

func = exp(2\*x);

ans = 1;

while (fabs(add) > eps && iter < 100) {

add = (pow(2\*x, iter))/factorial(iter);

iter++;

ans = ans + add;

}

printf("| %.3f | %.18lf | %.18lf | %d |\n", x,ans , func, iter);

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

x += (r - l) / n;

}

return 0;

}

# Входные данные

Единственная строка содержит два целых числа N (0≤N≤100) – число разбиений отрезка на равные части, K (0≤K≤16) — коэффициент для вычисления точности формулы Тейлора.

# Выходные данные

Программа должна вывести значение машинного эпсилон, а затем N+1 строку.

В каждой строке должно быть значение x, для которого вычисляется функция, число A1 — значение, вычисленное с помощью формулы Тейлора, A2 – значение, вычисленное с помощью встроенных функций языка, i – количество итерация, требуемых для вычисления, и Δ – разница значений A1 и A2 по модулю. A1, A2 и Δ должны быть выведены с точностью K знаков после запятой.

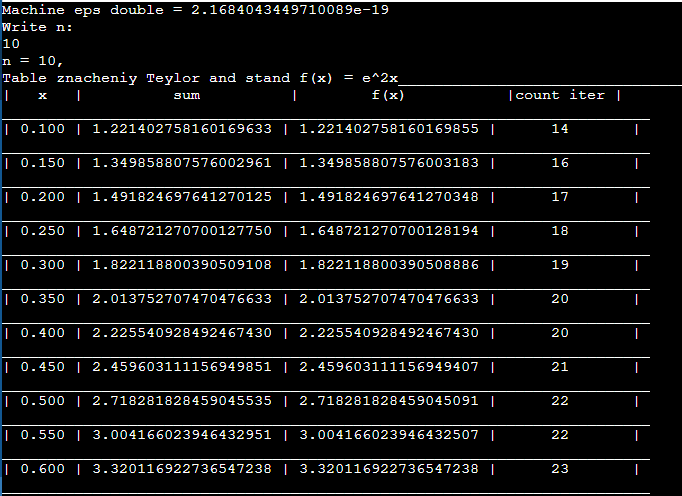
# Протокол исполнения и тесты

## Тест №1

Ввод:

10

Вывод:

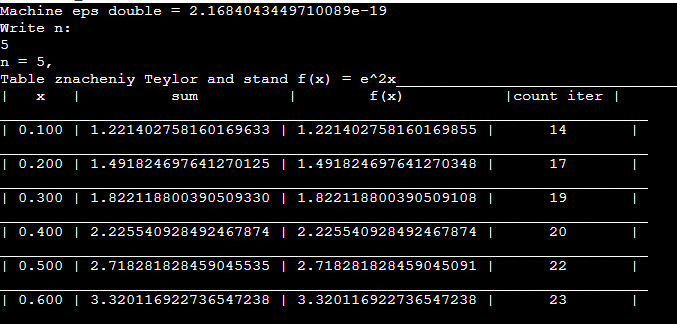


## Тест №2

Ввод:

5

Вывод:

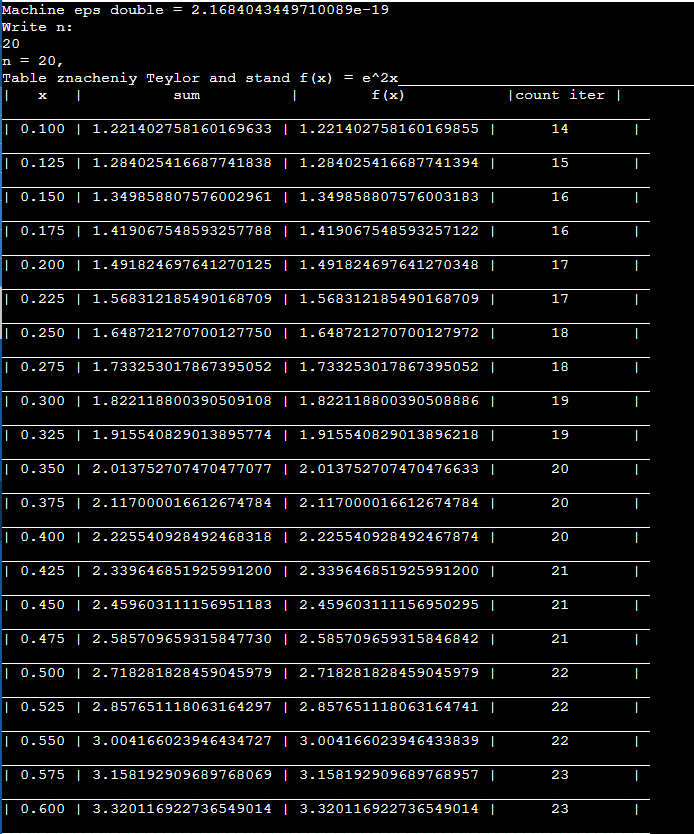


## Тест №3

Ввод:

20

Вывод:

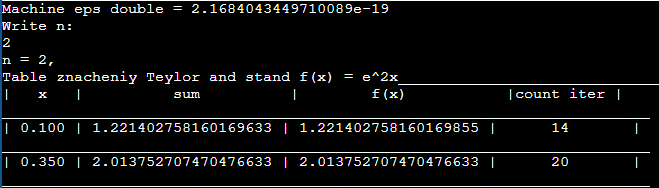


## Тест №4

Ввод:

2

Вывод:



# Вывод

В работе описано определение машинного эпсилон, приведены его значения для разных переменных языка Си, описана формула Тейлора и составлен алгоритм реализации вычисления значения функции с заданной точностью для заданного числа точек на отрезке. На основе алгоритма составлена программа на языке Си, проведено её тестирование на различных тестах, составлен протокол исполнения программы. В целом, работа понравилась. Приятно применять знания из других областей для решения какой-либо задачи по программированию.

# Список литературы

1. Машинный ноль – URL: <https://ru.wikipedia.org/wiki/Машинный_ноль>
2. Ряд Тейлора – URL: <https://ru.wikipedia.org/wiki/Ряд_Тейлора>