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»

Курсовой проект

по курсу «Вычислительные системы» I семестр

Задание 3

«Вещественный тип. Приближенные вычисления. Табулирование функций»

|  |  |
| --- | --- |
| Группа: | М8О-110Б-21 |
| Студент: | Агеева Алиса Ивановна |
| Преподаватель: |  |
| Оценка: |  |
| Дата: | 19.12.2021 |

# 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 27:

Ряд Тейлора

a = 0.1

b = 0.6

C:\Users\Администратор\Downloads\2021-12-19_18-27-47.png

# 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.

𝑘

∑

𝑛=0

𝑓(𝑛)(𝑎)

𝑛!

(𝑥 − 𝑎)𝑛 = 𝑓(𝑎) + 𝑓(1)(𝑎)(𝑥 − 𝑎) +

𝑓(2)(𝑎) 2!

(𝑥 − 𝑎)2+. . . +

𝑓(𝑘)(𝑎)

𝑘!

(𝑥 − 𝑎)𝑘

**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 за O(log(1016))~O(1) Для каждой N+1 строки нужно просуммировать i членов формулы Тейлора, пока |A1-A2|> ε. Проще всего

задать рекуррентное соотношение между членами: 𝑇𝑖+1

= (−1) 𝑇𝑖𝑥2

𝑛(𝑛−1)

где, Ti –

значение i-того члена формулы Тейлора, n – счётчик факториала, x – значение для которого проводятся вычисления. В общем случае сумма может быть вычислена за O(log k). Асимптотика решения O(1 + (n+1) log k) ~ O(n log k)

# Описание программы

## Использованные в программе переменные

|  |  |  |
| --- | --- | --- |
| Название переменной | Тип переменной | Смысл переменной |
| n | int | То самое число N, на которое нужно разбить отрезок |
| k | int | То самое число K, используемое для вычисления точности. Так же обозначает, что вывод должен быть с точность до K знаков после запятой |
| e0 | double | То самое машинное эпсилон. В случае с double ε =2.20 \* 10-16 |
| d | double | Формально разница между предыдущим значением из отрезка и следующим, если отрезок разбит на n равных частей. |
| x | double | Переменная, для которой будем производить вычисления |
| ans1 | double | То самое значение A1, вычисленное с помощью формулы Тейлора |
| ans2 | double | То самое значение A2, вычисленное с помощью встроенных функций языка |
| count | double | Счётчик члена формулы Тейлора. Так как для sin(x) все члены нечётные, то будем пропускать чётные |
| cur | double | То самое значение Ti. Текущие значение члена формулы Тейлора. При чём T0=x |
| step | int | Число итераций, которые затрачены на вычисление A2 |

**Использованные в программе функции**

|  |  |  |
| --- | --- | --- |
| Название функции | Тип возвращаемой переменной | Смысл функции |
| do\_nothing | double | Копирует значение в память, где double выделяется 64 бита, а не 80 бит |

# Программа

#include <math.h>

#include <stdio.h>

double do\_nothing(double x) {

return x;

}

int main()

{

int n, k;

printf("Задайте на сколько частей делить отрезок и количесвтво цифр после знака: ");

scanf("%d %d", &n, &k);

double e0 = 1.0, d = 0.5 / (double)n;

while (do\_nothing(1.0 + e0 / 2.0) > 1.0) {

e0 = e0 / 2.0;

}

printf("Machine epsilon equals %.8e\n", e0);

printf("--------------------------------------\n");

double x = 0.1, ans1, ans2, count, cur, cur0;

int step;

printf("|x | сумма ряда | функция | i | разница A1 A2|");

printf("\n================================================\n");

for (int i = 0; i <= n; i++) {

ans1 = 0;

ans2 = exp(cos(x))\*cos(sin(x));

step = 0;

count = 1.0;

cur = 1.0;

double copy = x;

if (copy == 0.1) {

step++;

ans1 = 1;

} else if (step == 0) {

ans1 = 1;

}

while ((e0 < fabs(ans1)) && (cur < 100)) {

cur \*=count;

cur0 = (cos(count\*x)) / cur;

ans1 += cur0;

step++; count++;

}

printf("%.2f | %.\*f | %.\*f | %d | %.\*f\n", x, k+1, ans1, k+1, ans2, step, k+1, fabs(ans2-

ans1));

x += d;

}

printf("================================================\n");

return 0;

}

# Входные и выходные данные

## 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.

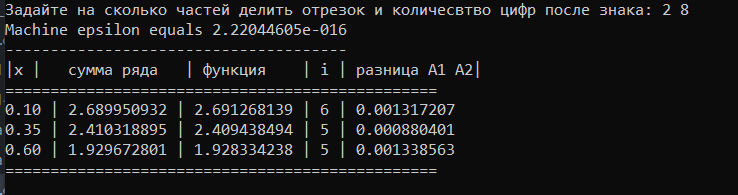
## Выходные данные

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

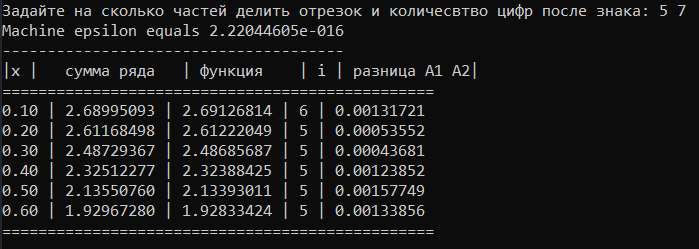
В каждой строке должно быть значение x, для которого вычисляется функция, число A1 — значение, вычисленное с помощью формулы Тейлора, A2 – значение, вычисленное с помощью встроенных функций языка, i – количество итерация, требуемых для вычисления, и Δ – разница значений A1 и A2 по модулю. A1, A2 и Δ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

**Протокол исполнения и тесты**

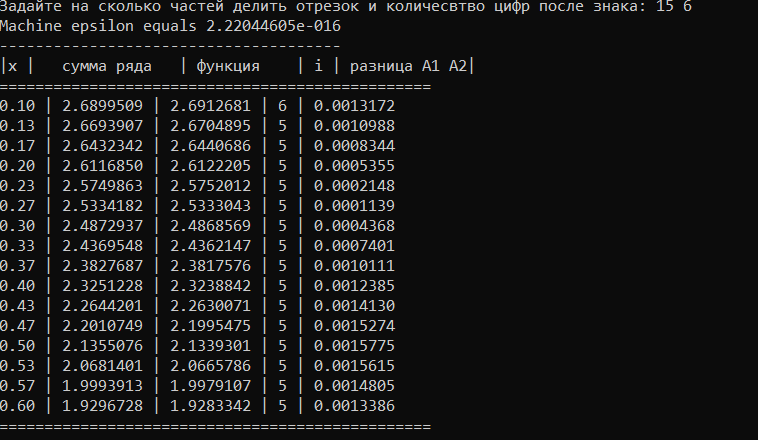
**Тест 1.**

****

**Тест 2.**



**Тест 3**



**Тест 4**

# C:\Users\Администратор\Downloads\2021-12-20_13-09-30.png

# Вывод

В работе описано определение машинного эпсилон, приведены его значения ля 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.

# Список литературы

1. Машинный ноль [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/Машинный_ноль>
2. Ряд Тейлора [Электронный ресурс] – URL: https://ru.wikipedia.org/wiki/Ряд\_Тейло

# 