МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Факультет компьютерных наук и прикладной математики

Кафедра вычислительной математики и программирования

**КУРСОВОЙ ПРОЕКТ**

**по курсу**

**«Вычислительные системы»**

Задание 3

«Вещественный тип. Приближённые вычисления. Табулирование функций»

**Выполнил:**

студент группы М8О-108Б-22

Формалёв Александр Сергеевич

**Преподаватель:**

Сахарин Никита Александрович

Дата:

Подпись:

Москва, 2022

**Цель работы**

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.

**Задание**

**Функция:**

**Ряд:**

**Отрезок:** [0.1;0.6]

**Теоретическая часть.**

**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 минимальное число, удовлетворяющее неравенству 1 + ε > 1. **Машинный ноль** — числовое значение с таким отрицательным порядком, которое воспринимается машиной как ноль.

**Формула Тейлора:** Пусть функция определена в окрестности точки и имеет в этой окрестности производные до порядка включительно, и пусть существует . Тогда

.

Многочлен – многочлен Тейлора.

При – формула Маклорена.

**Код программы**

#include <stdio.h>

#include <math.h>

#define MAX\_ITER (100)

typedef double dbl;

dbl Taylor\_exp\_x(dbl x);

void table(dbl a, dbl b, int n);

dbl epsilon();

int main(){

int n;

printf("The number of points: ");

scanf("%d", &n);

printf("Machin epsilon for long double = %.10lg\n", epsilon());

table(0.1, 0.6, n);

return 0;

}

dbl epsilon(){

dbl eps = 1;

while(1 + eps / 2.0 != 1){

eps /= 2.0;

}

return eps;

}

dbl Taylor\_exp\_x(dbl x){

dbl result = 1, term, numerator = 1, denominator = 1;

for (int i = 1; i < MAX\_ITER; ++i){

numerator \*= x;

denominator \*= i;

term = numerator / denominator;

if (result + term == result)

break;

result += term;

}

return result;

}

void table(dbl a, dbl b, int n){

printf(" Arg | C-function | Taylor\n");

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

dbl delta = (b - a) / n, x = a;

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

x = a + delta \* i;

printf("%.5lf | %.14lf | %.14lf\n", x, exp(2 \* x), Taylor\_exp\_x(2 \* x));

}

}

**Входные данные**

На вход поступает число: количество точек, значения функции в которых должна вычислить программа двумя способами.

**Выходные данные**

Программа должна вывести значение машинного эпсилон, а затем таблицу значений функции с их аргументами. В каждой строке находится значение аргумента, для которого вычисляется функция, затем значение, вычисленное с помощью формулы Тейлора, и, наконец, значение, вычисленное с помощью встроенных функций языка.

**Протокол исполнения**

The number of points: 20

Machin epsilon for long double = 2.220446049e-16

Arg | C-function | Taylor

----------|-------------------------|------------------------

0.10000 | 1.22140275816017 | 1.22140275816017

0.12500 | 1.28402541668774 | 1.28402541668774

0.15000 | 1.34985880757600 | 1.34985880757600

0.17500 | 1.41906754859326 | 1.41906754859326

0.20000 | 1.49182469764127 | 1.49182469764127

0.22500 | 1.56831218549017 | 1.56831218549017

0.25000 | 1.64872127070013 | 1.64872127070013

0.27500 | 1.73325301786740 | 1.73325301786740

0.30000 | 1.82211880039051 | 1.82211880039051

0.32500 | 1.91554082901390 | 1.91554082901390

0.35000 | 2.01375270747048 | 2.01375270747048

0.37500 | 2.11700001661267 | 2.11700001661267

0.40000 | 2.22554092849247 | 2.22554092849247

0.42500 | 2.33964685192599 | 2.33964685192599

0.45000 | 2.45960311115695 | 2.45960311115695

0.47500 | 2.58570965931585 | 2.58570965931585

0.50000 | 2.71828182845905 | 2.71828182845905

0.52500 | 2.85765111806316 | 2.85765111806316

0.55000 | 3.00416602394643 | 3.00416602394643

0.57500 | 3.15819290968977 | 3.15819290968977

0.60000 | 3.32011692273655 | 3.32011692273655

**Вывод**

В ходе выполнения курсового проекта были получены знания о реализации вещественных типов данных в языке Си. Также были изучены некоторые математические аспекты, такие как формула Тейлора и ряд Маклорена.