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 «Фундаментальная информатика»
I семестр
Задание 3
«Вещественный тип. Приближенные вычисления. Табулирование
функций»

Группа	М8О-109Б-22
Студент	Фомин И.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 $\varepsilon * 10^k$, 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:

$$1+2\frac{x}{2}+\ldots+\frac{n^2+1}{n!}(\frac{x}{2})^n$$

Функция:

$$(\frac{x^2}{4}+\frac{x}{2}+1)e^{\frac{x}{2}}$$

Значения a 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.

$$\sum_{n=0}^{k} \frac{f^{(n)}(a)}{n!}(x-a)^n = f(a)+f^{(1)}(a)(x-a)+\frac{f^{(2)}(a)}{2!}(x-a)^2+\ldots+\frac{f^{(k)}(a)}{k!}(x-a)^k$$

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 $1 + \varepsilon = 1$. 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 $|A_1-A_2| > \varepsilon$. 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	int64_t	То самое число N, на которое нужно разбить отрезок
k	int	То самое число K, используемое для вычисления точности.
FLT_EPSILON	float	То самое машинное эпсилон. 1.192092896e-07F
step	long double	Формально разница между предыдущим значением из отрезка и следующим, если отрезок разбит на n равных частей.
currentX	long double	Переменная, для которой будем производить вычисления
getTaylorSeries (currentX, i)	double	То самое значение A_1, вычисленное с помощью формулы Тейлора
func(currentX)	double	То самое значение A_2, вычисленное с помощью встроенных функций языка
i	double	Счётчик члена формулы Тейлора + кол-во итераций

Исходный код программы:

```
#include <stdio.h>
#include <float.h>
#include <stdint.h>
#include <math.h>

int64_t factorial(int64_t n) {
    int64_t res = 1;

    for (int64_t i = 1; i <= n; ++i)
        res *= i;

    return res;
}

long double func(long double x) {
    return (x * x / 4 + x / 2 + 1) * expl(x / 2);
}

long double getTaylorSeries(long double x, int64_t n) {
    long double res = 0;
    for (int64_t i = 0; i <= n; ++i)
        res += (((long double) (i * i + 1)) / ((long double) factorial(i)))
* powl(x / 2, (long double) i);

    return res;
}

int main() {
    long double sum = 0.0;
    long double a = 0.1;
    long double b = 0.6;

    int64_t n;
    scanf_s("%lld", &n);
    printf("N = %lld\n", n);
    printf("Machine epsilon is equal to: %g\n", LDBL_EPSILON);

    printf("Table for values of Taylor series and of base function\n");

printf("
_____\n");
    printf("|   x   |          sum       |        f(x)        |number of
iterations |\n");

printf("
_____\n");

    long double currentX;
    long double step = (b - a) / (long double) n;
    for (int64_t i = 1; i <= n; ++i) {
        currentX = a + step * (long double) i;
        sum = getTaylorSeries(currentX, i);

        printf("| %.3Lf | %.16Lf | %.16Lf |       %lld       |\n", currentX,
sum, func(currentX), i);

printf("
_____\n");
```

```
        if (fabsl(func(currentX) - sum) < LDBL_EPSILON) break;
    }

    return 0;
}
```

Входные данные

Единственная строка содержит одно целое число N (0≤N≤100) – число разбиений отрезка на равные части

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 A_1 — значение, вычисленное с помощью формулы Тейлора, A_2 – значение, вычисленное с помощью встроенных функций языка, i – количество итерация, требуемых для вычисления, и Δ – разница значений A_1 и A_2 по модулю. A_1, A_2 и Δ должны быть выведены с точностью 16 знаков после запятой.

Протокол исполнения и тесты

Тест №1

Ввод:

2

Вывод:

```
2
N = 2
Machine epsilon is equal to: 2.22045e-16
Table for values of Taylor series and of base function
-------------------------------------------------------------------------
|   x   |          sum       |        f(x)        |number of iterations |
-------------------------------------------------------------------------
| 0.350 | 1.3500000000000001 | 1.4361962199032741 |      1       |
-------------------------------------------------------------------------
| 0.600 | 1.8250000000000002 | 1.8763037425306446 |      2       |
-------------------------------------------------------------------------
```

Process finished with exit code 0

Тест №2

Ввод:
200

Вывод:

```
200
N = 200
Machine epsilon is equal to: 2.22045e-16
Table for values of Taylor series and of base function
-------------------------------------------------------------------------
|   x   |          sum       |        f(x)        |number of iterations |
-------------------------------------------------------------------------
| 0.103 | 1.1025000000000000 | 1.1092957226814248 |      1       |
-------------------------------------------------------------------------
| 0.105 | 1.1118906250000000 | 1.1121372655243895 |      2       |
-------------------------------------------------------------------------
| 0.108 | 1.1149814680989583 | 1.1149874787883696 |      3       |
-------------------------------------------------------------------------
| 0.110 | 1.1178462733593750 | 1.1178463838420398 |      4       |
-------------------------------------------------------------------------
| 0.113 | 1.1207140004534721 | 1.1207140020991240 |      5       |
-------------------------------------------------------------------------
| 0.115 | 1.1235903549976711 | 1.1235903550184816 |      6       |
-------------------------------------------------------------------------
| 0.118 | 1.1264754641039618 | 1.1264754641041927 |      7       |
-------------------------------------------------------------------------
| 0.120 | 1.1293693509056417 | 1.1293693509056446 |      8       |
-------------------------------------------------------------------------
| 0.122 | 1.1322720370176180 | 1.1322720370176180 |      9       |
-------------------------------------------------------------------------
```

Тест №3

Ввод:

100000

Вывод:

```
100000
N = 100000
Machine epsilon is equal to: 2.22045e-16
Table for values of Taylor series and of base function
```

x	sum	f(x)	number of iterations
0.100	1.1000049999999999	1.1064684861056084	1
0.100	1.1062612500625000	1.1064741433099630	2
0.100	1.1064753022380216	1.1064798005488288	3
0.100	1.1064853892343978	1.1064854578222068	4
0.100	1.1064911143181122	1.1064911151300965	5
0.100	1.1064967724646682	1.1064967724724986	6
0.100	1.1065024298493489	1.1065024298494124	7
0.100	1.1065080872608386	1.1065080872608390	8
0.100	1.1065137447067783	1.1065137447067783	9

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