Московский авиационный институт
(национальный исследовательский университет)

Факультет № 8 «Прикладная математика и информатика»

Кафедра 806 «Вычислительная математика и программирование»

КУРСОВОЙ ПРОЕКТ

по дисциплине «Вычислительные системы»

1 семестр

на тему “Схема лабораторной вычислительной системы”

Студент:	Соколов Д. В.
Группа:	М8О-107Б-20
Преподаватель:	Найдёнов И. Е.
Подпись:	
Оценка:	

Москва, 2020

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 5:

функция: $2(\cos^2 x - 1)$

Ряд Тейлора: $-\frac{4x^2}{2}+\frac{16x^4}{24}+\ldots+(-1)^n\frac{(2x)^{2n}}{(2n)!}$

отрезок [a;b] → [0.0; 0.5]

Теоретическая часть:

<u>Машинный ноль (машинный нуль)</u> — это очень маленькое число , которое воспринимается ЭВМ как ноль (из-за недостатка точности вещественных чисел)
<u>Машинный эпсилон</u> — числовое значение, меньше которого невозможно задавать относительную точность для любого алгоритма, возвращающего вещественные числа.(Эпсилон немного больше машинного нуля , позволяя сравнивать вещественные числа в ЭВМ)
Обычно эпсилон находится по формуле 1 + ε = 1 , поэтому применяя определение предела из математического анализа для двух отличных от нуля переменных , эти числа будут равны в том случаи, если модуль их разности не превосходит эпсилон , где ε > 0.
<u>Ряд Тейлора</u> — это разложение функции в бесконечную сумму степенных функций Формула Тейлора используется для приближенного вычисления сложный (даже тригонометрических) функций.

Алгоритм выполнения задачи:

С самого начала нужно найти эпсилон ЭВМ , на которой происходит выполнение программы. Для этого нужно приблизиться к машинному нулю :

```
float eps = 1.0f;
while (1.0f + eps / 2.0f > 1.0f) {
    eps /= 2.0f;
 }
```

Цикл while будет выполнять деление эпсилон на 2 , пока 1 + ε/2 не станет равняться 1, т.е. ε/2 будет считаться машинным нулём , а следовательно ε это и будет число приблежённое к нулю , но немного больше , а значит ε — это машинный эпсилон. Такой поиск эпсилон выполняется за $O(\log(10^{16}))\sim O(1)$. С помощью библиотеки <math.h> будем находить точное значиние от данной функции, а приближенное значение по формуле Тейлора. Чтобы приближенное значение отличалось от точного не больше чем на эпсилон , воспользуемся определением предела , т.е. сумма Тейлора стремится к точному значению , вычисленной по математическим формулам, и получим $|yt - y| < \varepsilon$, где $y = f(x)$, а yt - это сумма ряда.

Программа и вывод тестирования:

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define N 20

int fact(int n)
{
   if (n == 0) {
      return 1;
   }
   while (n != 0) {
      return fact(n - 1) * n;
   }
}

int main (void)
{
   float y, yt = 0.0f;
   float x = 0.0f;
   float eps = 1.0f;
   while (1.0f + eps / 2.0f > 1.0f) {
```

```
        eps /= 2.0f;
    }
    printf("epsilon = %.8f\n", eps);
    printf("-----------------------------------------------\n");
    printf("|  x  |y = 2(cosx ^ 2 - 1)|    y = taylor:    |n|\n");
    printf("-----------------------------------------------\n");
    int i, n = 1;
    for (i = 0; i <= N; i++) {
        y = 2 * (cos(x)*cos(x) - 1);
        while (fabs(yt - y) > eps) {
            yt = yt + pow(-1, n) * pow((2*x), 2*n) / (fact(2*n));
            n++;
        }
        if (y >= 0.0) {
            printf("|%.3f|     %.8f    |     %.8f    |%d|\n", x, y, yt, n);
        } else {
            printf("|%.3f|    %.8f    |    %.8f    |%d|\n", x, y, yt, n);
        }
        n = 1;
        yt = 0.0f;
        x = x + 0.025f;
    }
    printf("-----------------------------------------------\n");
    return 0;
}
```

-Вывод тестирования-

```
epsilon = 0.00000012
-----------------------------------------------
|  x  |y = 2(cosx ^ 2 - 1)|    y = taylor:    |n|
-----------------------------------------------
|0.000|     0.00000000    |     0.00000000    |1|
|0.025|    -0.00124974    |    -0.00124974    |3|
|0.050|    -0.00499584    |    -0.00499583    |3|
|0.075|    -0.01122892    |    -0.01122891    |3|
|0.100|    -0.01993342    |    -0.01993334    |3|
|0.125|    -0.03108758    |    -0.03108758    |4|
|0.150|    -0.04466351    |    -0.04466351    |4|
|0.175|    -0.06062730    |    -0.06062730    |4|
|0.200|    -0.07893902    |    -0.07893904    |4|
|0.225|    -0.09955292    |    -0.09955296    |4|
|0.250|    -0.12241746    |    -0.12241756    |4|
|0.275|    -0.14747551    |    -0.14747551    |5|
|0.300|    -0.17466444    |    -0.17466444    |5|
```

0.325	**-0.20391627**	**-0.20391627**	**5**
0.350	**-0.23515788**	**-0.23515788**	**5**
0.375	**-0.26831120**	**-0.26831120**	**5**
0.400	**-0.30329338**	**-0.30329335**	**5**
0.425	**-0.34001696**	**-0.34001690**	**5**
0.450	**-0.37839016**	**-0.37839007**	**5**
0.475	**-0.41831705**	**-0.41831708**	**6**
0.500	**-0.45969778**	**-0.45969778**	**6**

--

Вывод:

В ходе курсового проекта изучено представление вещественных чисел в ЭВМ , а также приобретины навыки сравнение их на языке Си. Пришлось находить машинный эпсилон для любого ПК , и использовать его в вычислении приближенных значений. С помощью знаний математического анализа находил сумму ряда Тейлора и его предел. В работе составлена программа на языке Си на основе алгоритма , описанного в проекте.

Список литературы:

Машинный ноль [Электронный ресурс] – URL:
https://ru.wikipedia.org/wiki/Машинный_ноль
Ряд Тейлора [Электронный ресурс] – URL:
https://ru.wikipedia.org/wiki/Ряд_Тейлора