**Московский Авиационный Институт**

**(Национально Исследовательский Университет)**

**Факультет информационных технологий и прикладной**

**математики**

**Кафедра вычислительной математики и программирования**

**Курсовая проект по курсу**

**«Вычислительные системы»**

**Первый семестр**

**Задание 3**

«Вещественный тип. Приближенные вычисления»

|  |  |
| --- | --- |
| Группа: | М8О-110Б-22 |
| Студент: | Филиппов Владимир Михайлович |
| Преподаватель: | Днепров Иван Сергеевич |
| Оценка: |  |
| Дата: | 10.12.2021 |

**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 (Вариант 20)**

**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.

**Машинный ноль** — числовое значение с таким отрицательным порядком, которое воспринимается машиной как ноль.

**Код программы.**

#include <stdio.h>

#include <math.h>

long double Abs(long double num) {

return num > 0 ? num : -num;

}

long double Pow(long double num, long double base) {

long double res = num;

for (int i = 0; i < 2; i++) {

res \*= base;

}

return res;

}

long double MachineEpsilon() {

long double e = 1.0l;

while (1.0l + e \* 0.5l > 1.0l) {

e \*= 0.5l;

}

return e;

}

int main(void) {

long double e = MachineEpsilon();

long double b = 0.5, step, res, temp, num;

int counter = 0, iter;

printf("Print the iterations: ");

scanf("%d", &iter);

putchar('\n');

step = b / iter;

printf("Machine epsilon for long double = %.8Le\n", e);

printf("Taylor series values table for f(x) = arctan(x)\n");

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

printf("| x \t| sum of line\t\t| function\t\t| iter |\n");

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

for (long double i = 0; i <= b; i += step) {

num = i;

for (int n = 0; n < 999; n++) {

if (n % 2 == 0) {

temp = (num / (2 \* n + 1));

} else {

temp = -(num / (2 \* n + 1));

}

res += temp;

counter += 1;

num = Pow(num, i);

if (Abs(temp) < e){

break;

}

}

printf("| %.3Lf\t| %.20Lf| %.20Lf| %d\t|\n", i, res, atanl(i), counter);

res = 0;

counter = 0;

}

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

return 0;

}

**Входные данные**

Единственная строка содержит два целых числа 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 A1 — значение, вычисленное с помощью формулы Тейлора, A2 – значение, вычисленное с помощью встроенных функций языка, i – количество итераций. A1, A2 должны быть выведены с приемлемой точностью.

**Протокол исполнения и тесты:**

Тест #1

Ввод: 5

Print the iterations: 5

Machine epsilon for long double = 1.08420217e-19

Taylor series values table for f(x) = arctan(x)

---------------------------------------------------------------

| x | sum of line | function | iter |

-------+-----------------------+-----------------------+-------

| 0.000 | 0.00000000000000000000| 0.00000000000000000000| 1 |

| 0.100 | 0.09966865249116202737| 0.09966865249116202738| 10 |

| 0.200 | 0.19739555984988075839| 0.19739555984988075837| 14 |

| 0.300 | 0.29145679447786709200| 0.29145679447786709200| 18 |

| 0.400 | 0.38050637711236488629| 0.38050637711236488632| 23 |

| 0.500 | 0.46364760900080611615| 0.46364760900080611620| 30 |

----------------------------------------------------------------

**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.

При этом каждое значение вычисляется до 100 итераций, а дополнительные переменные помогают нам не считать заново каждое значение.