МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

“Московский Авиационный Институт

(Национальный Исследовательский Университет)”

Факультет информационных технологий и прикладной математики

Кафедра 806 “Вычислительная математика и программирование”

Курсовая работа

по курсу “Вычислительные системы”

1 семестр

Задание 3. Вещественный тип. Приближенные вычисления. Табулирование функций

Студент: Немкова А.Р.

Группа: М8О-108Б-22

Руководитель: Сахарин Н.А.

Дата: 28.12.12

Оценка:

Подпись:

Москва, 2022

**Содержание**

ЗАДАЧА …………………………………………………………………...…. 3

ВАРИАНТ …………………………………………………………………..... 3

ОБЩИЙ МЕТОД РЕШЕНИЯ ……………………………………………… 3

ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ ……………………………………. 3

ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ……………………………………. 4

ОПИСАНИЕ ПРОГРАММЫ ……………………………………………….. 4

ПРОТОКОЛ ………………………………………………………………….. 6

ВЫВОДЫ …………………………………………………………………… 10

**1. Задача**

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\*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.

**2. Вариант**

| № | Ряд | a | b | Функция |
| --- | --- | --- | --- | --- |
| 14 |  | 0,1 | 0,6 |  |

**3. Общий метод решения**

Вычисление значения функции в некоторой точке на отрезке от 0,1 до 0,6 двумя способами.

1 способ - использование программных средств, встроенных в стандартную математическую библиотеку языка Си “math.c”.

2 способ - при помощи ряда Тейлора.

**4. Общие сведения о программе**

Аппаратное обеспечение: домашний ноутбук

Операционная система: Linux Ubuntu, версия 22.04.1 LTS

Язык и система программирования: С, GNU

Местонахождение файлов: /home/anastasia

Компиляция программы: gcc -lm kp3.c

Вызов программы: ./a.out

**5. Функциональное назначение**

Программа предназначена для высокоточного вычисления вещественного значения трансцедентных функций в алгебраической форме с использованием ряда Тейлора и при помощи встроенных программных функций библиотеки языка Си.

**6. Описание программы**

Программа вычисляет значение функции в данной точке при помощи разложения по ряду Тейлора и с использованием программных средств языка программирования СИ. Ряд Тейлора мы преобразуем в функцию, которая вычисляет слагаемые ряда. Далее мы складываем полученные слагаемые, пока их количество не превысит 100 или значение одного из них не станет совсем мало (меньше ε\*л по модулю). В конце мы выводим таблицу с значением аргумента, значением функции, вычисленным с помощью ряда Тейлора и с использованием программной библиотеки, и номером итерации.

Таблица 1. Описание функций программы

| Функция | Входные аргументы | Описание |
| --- | --- | --- |
| machine\_eps | - | Функция для подсчета машинного ε. Сравниваем 1+ε с 1. Последнее число, при стремлении ε к нулю, при котором 1+ε > 1 и будет машинным ε |
| Abs | long double a | Функция для вычисления модуля входного числа |
| Pow | long double x, long double n, long double i | Функция, вычисляющая число , при n = 0, получаем 1; при n = 1 получаем x; при n > 1 получаем x\*i (число i равно x) |
| func | long double x | Вычисляет значение входной функции при помощи встроенной библиотеки “math.c” |
| teilor\_member | long double x, int n, long double i | Выводит n-ое слагаемое из ряда Тейлора |

Таблица 2. Описание переменных

| Переменная | Значение |
| --- | --- |
| long double eps | Машинный эпсилон |
| const long double k | Эмпирический коэффициент для эпсилон |
| int MAX\_ITER | Максимальное число итераций |
| long double b, а | Границы отрезка |
| long double step | Количество отрезков |
| long double iter | Количество итераций |
| long double result | Сумма ряда |
| long double cur\_number | n-ое слагаемое ряда |
| long double x | Значение аргумента функции |
| long double i | Переменная, обозначающая тоски, на которые разбивается отрезок |
| int count | Количество итерации при вычислении функции при помощи ряда Тейлора (количество слагаемых) |
| int n | Текущая итерация |

**7. Протокол**

Код программы:

#include <stdio.h>

#include <math.h>

const long double k = 10e2;

int MAX\_ITER = 100;

long double machine\_eps() {

long double eps = 1.0;

while ((1.0 + eps) > 1.0) {

eps \*= 0.5;

}

return eps;

}

long double Abs(long double a) {

return a >= 0 ? a : -a;

}

long double func(long double x) {

return (2 \* x - 3) / ((x - 1) \* (x - 1));

}

long double Pow(long double x, long double n, long double i) {

if (n == 0) {

return 1;

}

else if (n == 1) {

return x;

}

else if (n > 1) {

return x \* i;

}

return 0;

}

long double teilor\_member(long double x, int n, long double i){

long double sum = -(n + 3) \* Pow(x, n, i);

return sum;

}

int main() {

long double eps = machine\_eps();

long double b = 0.6, step, iter, result, cur\_number, x;

long double a = 0.1;

int count = 0;

printf("Print the iterations: ");

scanf("%Lf", &iter);

putchar('\n');

step = (b - a) / iter;

printf("Machine epsilon for long double = %.20Lf\n", eps);

printf("Taylor series values table for f(x) = (2x-3)/(x-1)^2\n");

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

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

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

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

x = i;

for (int n = 0; n <= MAX\_ITER ; n++) {

cur\_number = teilor\_member(x, n, i);

result += cur\_number;

count ++;

if (n > 1) {

x \*= i;

}

if (Abs(cur\_number) < eps\*k) {

break;

}

}

if (Abs((int)result) >= 10) {

printf("| %.3Lf\t| %.20Lf| %.20Lf| %d |\n", i, result, func(i), count);

} else {

printf("| %.3Lf\t| %.20Lf | %.20Lf | %d |\n", i, result, func(i), count);

}

result = 0;

count = 0;

}

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

return 0;

}

Выходные данные:

anastasia@anastasia-VirtualBox:~$ gcc -lm kp3.c

anastasia@anastasia-VirtualBox:~$ ./a.out

Print the iterations: 3

Machine epsilon for long double = 0.00000000000000000005

Taylor series values table for f(x) = (2x-3)/(x-1)^2

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

| x | sum of line | function | iter |

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

| 0.100 | -3.45679012345679014992 | -3.45679012345679015231 | 19 |

| 0.267 | -4.58677685950413219391 | -4.58677685950413219869 | 33 |

| 0.433 | -6.64359861591695476666 | -6.64359861591695479398 | 51 |

| 0.600 | -11.24999999999999898085| -11.24999999999999902855| 84 |

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

anastasia@anastasia-VirtualBox:~$ ./a.out

Print the iterations: 8

Machine epsilon for long double = 0.00000000000000000005

Taylor series values table for f(x) = (2x-3)/(x-1)^2

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

| x | sum of line | function | iter |

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

| 0.100 | -3.45679012345679014992 | -3.45679012345679015231 | 19 |

| 0.163 | -3.81376698596569392802 | -3.81376698596569393149 | 24 |

| 0.225 | -4.24557752341311132492 | -4.24557752341311133186 | 29 |

| 0.287 | -4.77685441674361335176 | -4.77685441674361337345 | 34 |

| 0.350 | -5.44378698224852059602 | -5.44378698224852060989 | 41 |

| 0.412 | -6.30149388863739226985 | -6.30149388863739230065 | 48 |

| 0.475 | -7.43764172335600870544 | -7.43764172335600874837 | 57 |

| 0.537 | -8.99926953981007976026 | -8.99926953981007979669 | 69 |

| 0.600 | -11.24999999999999898085| -11.24999999999999902855| 84 |

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

anastasia@anastasia-VirtualBox:~$ ./a.out

Print the iterations: 15

Machine epsilon for long double = 0.00000000000000000005

Taylor series values table for f(x) = (2x-3)/(x-1)^2

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

| x | sum of line | function | iter |

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

| 0.100 | -3.45679012345679014992 | -3.45679012345679015231 | 19 |

| 0.133 | -3.63905325443786984178 | -3.63905325443786984373 | 22 |

| 0.167 | -3.84000000000000000491 | -3.84000000000000001185 | 24 |

| 0.200 | -4.06249999999999999610 | -4.06250000000000000043 | 27 |

| 0.233 | -4.31001890359168240161 | -4.31001890359168240551 | 30 |

| 0.267 | -4.58677685950413219435 | -4.58677685950413219869 | 33 |

| 0.300 | -4.89795918367346932434 | -4.89795918367346933258 | 36 |

| 0.333 | -5.24999999999999990069 | -5.24999999999999991760 | 39 |

| 0.367 | -5.65096952908587244315 | -5.65096952908587245789 | 43 |

| 0.400 | -6.11111111111111093060 | -6.11111111111111094708 | 47 |

| 0.433 | -6.64359861591695476666 | -6.64359861591695479528 | 51 |

| 0.467 | -7.26562499999999966823 | -7.26562499999999970249 | 56 |

| 0.500 | -7.99999999999999957326 | -7.99999999999999960188 | 62 |

| 0.533 | -8.87755102040816268787 | -8.87755102040816273298 | 68 |

| 0.567 | -9.94082840236686312956 | -9.94082840236686318940 | 75 |

| 0.600 | -11.24999999999999898085| -11.24999999999999902855| 84 |

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

**8. Вывод**

В ходе выполнения данной работы были получены навыки вычисления заданной функции при помощи разложения по ряду Тейлора и с помощью встроенной библиотеки СИ “math.c”, было изучено вычисление и использование машинного эпсилон. После составление таблицы значений заданной функции, можно увидеть, что значения различаются приблизительно после 16 знака после запятой. Это происходит из-за ограниченности разрядной сетки при представлении вещественных чисел, так как для данных чисел выделяется ограниченное количество памяти в компьютере, что приводит к тому, что в окрестностях границ данного диапазона возникают погрешности.

Вычисление трансцендентных функций при помощи формулы Тейлора не применяется на практике ввиду большой ресурсоемкости и значительной погрешности.