**Отчет по курсовой работе №3** по курсу\_1\_\_\_\_\_

|  |  |
| --- | --- |
|  | Студент группы М80-105Б-21 Жилов Андрей, № по списку 4  Контакты www, e-mail: [klzxrcn3692@outlook.com](mailto:klzxrcn3692@outlook.com)\_\_\_\_\_\_\_\_\_\_\_  Работа выполнена: «7» декабря 2021г.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Преподаватель: В. К. Титов каф. 806\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отчет сдан «7» декабря 2021 г., итоговая оценка \_\_\_\_\_\_\_\_\_  Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

1. **Тема:** Вещественный тип. Приближенные вычисления. Табулирование функций \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. **Цель работы:** Составить программу на Си, которая печатает таблицу значений элементарной функции, вычисленной двумя способами: по формуле Тейлора и с поммощью 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.
3. **Задание****:** (*Вариант №4*):

1. **Оборудование(лабораторное)**:

ЭВМ -, процессор -, имя узла сети - с ОП - ГБ,

НМД - ГБ, терминал- адрес -, принтер –

Другие устройства –

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 3500U, с ОП 8 ГБ, НМД 256 ГБ. Монитор 1920x1080 пикс.

Другие устройства –

1. **Программное обеспечение:**

Операционная система семейства -, наименование - версия –

интерпретатор команд – версия

Система программирования - версия –

Редактор текстов - версия –

Утилиты операционной системы –

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства GNU/Linux, наименование Kali версия 5.10.0

интерпретатор команд Bash версия 5.1.8(1)-release

Система программирования \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

редактор текстов emacs версия 27.1

Утилиты операционной системы ls, cat, date, cal, who, whoami, uname, ruptime, uptime, tty,hostname, pwd, cd,mkdir,

rmdir, cp, mv, rm, whatis, whereis, chmod, sudo, lsblk, df, free, ps, echo.

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

1. **Идея, метод, алгоритм**  решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок,

таблица] или формальные спецификации с пред- и постусловиями)

Сначала мы высчитываем машинный эпсилон(делим его на два и сравниваем 1+ эпсилон и 1, так пока 1+ эпсилон не будет настолько незначителен, что 1+эпсилон не больше 1)

Затем начинаем выводить таблицу значений функции для каждого x мы прочитываем значения через ряд Тейлора и через стандартную функцию языка Си

1. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

#include <stdio.h>

#include <math.h>

double my\_log(const double& epsilon,const double& x,int& n)

{

double p=x/2,S=p;

while(abs(p)>epsilon)

{

++n;

if(n>99)

break;

p=-p\*x\*(n-1)/(n\*2);

S+=p;

}

return S;

}

int main()

{

double x,S,f, epsilon= 1;

float a=-1.0, b = 1.0, h = (b-a)/10;

int n;

while (1+epsilon>1)

epsilon/=2;

printf("Computer epsilon: %.20f\n",epsilon);

printf("┌──────┬────────────────────────┬────────────────────────┬────┐\n");

printf("│ X │ Sum │ f(x) │ n │\n");

for(x=a;x<=b;x+=h)

{

n=1;

S=log(2)+my\_log(epsilon,x,n);

f=log(x+2);

printf("├──────┼────────────────────────┼────────────────────────┼────┤\n");

if(x>=0)

printf("│ %.2f │",x);

else

printf("│%.2f │",x);

printf(" %.20f │ %.20f │ %2d │\n",S,f,n);

}

x=1.0;

n=1;

S=log(2)+my\_log(epsilon,x,n);

f=log(x+2);

printf("├──────┼────────────────────────┼────────────────────────┼────┤\n");

printf("│ %.2f │ %.20f │ %.20f │ %d │\n",x,S,f,n);

printf("└──────┴────────────────────────┴────────────────────────┴────┘\n");

}

*Пункты 1-7 отчета составляются сторого до начала лабораторной работы.*

*Допущен к выполнению работы.*  **Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Распечатка протокола**  (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

┌──(kali㉿kali)-[/media/…/FFFFFFFf/лабы и дз фунд/mylab/curs3]

└─$ cat curs3.txt

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* Курсовая работа #3 \*

\* Вещественный тип. Приближенные \*

\* вычисления. Табулирование функций. \*

\* Выполнил студент группы \*

\* М8О-105Б-21 \*

\* Жилов Андрей Алексеевич \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

┌──(kali㉿kali)-[/media/…/FFFFFFFf/лабы и дз фунд/mylab/curs3]

└─$ cat kurs3.c

#include <stdio.h>

#include <math.h>

double my\_log(const double& epsilon,const double& x,int& n)

{

double p=x/2,S=p;

while(abs(p)>epsilon)

{

++n;

if(n>99)

break;

p=-p\*x\*(n-1)/(n\*2);

S+=p;

}

return S;

}

int main()

{

double x,S,f, epsilon= 1;

float a=-1.0, b = 1.0, h = (b-a)/10;

int n;

while (1+epsilon>1)

epsilon/=2;

printf("Computer epsilon: %.20f\n",epsilon);

printf("┌──────┬────────────────────────┬────────────────────────┬────┐\n");

printf("│ X │ Sum │ f(x) │ n │\n");

for(x=a;x<=b;x+=h)

{

n=1;

S=log(2)+my\_log(epsilon,x,n);

f=log(x+2);

printf("├──────┼────────────────────────┼────────────────────────┼────┤\n");

if(x>=0)

printf("│ %.2f │",x);

else

printf("│%.2f │",x);

printf(" %.20f │ %.20f │ %2d │\n",S,f,n);

}

x=1.0;

n=1;

S=log(2)+my\_log(epsilon,x,n);

f=log(x+2);

printf("├──────┼────────────────────────┼────────────────────────┼────┤\n");

printf("│ %.2f │ %.20f │ %.20f │ %d │\n",x,S,f,n);

printf("└──────┴────────────────────────┴────────────────────────┴────┘\n");

}

┌──(kali㉿kali)-[/media/…/FFFFFFFf/лабы и дз фунд/mylab/curs3]

└─$ g++ kurs3.c

┌──(kali㉿kali)-[/media/…/FFFFFFFf/лабы и дз фунд/mylab/curs3]

└─$ ./a.out

Computer epsilon: 0.00000000000000011102

┌──────┬────────────────────────┬────────────────────────┬────┐

│ X │ Sum │ f(x) │ n │

├──────┼────────────────────────┼────────────────────────┼────┤

│-1.00 │ 0.00000000000000022204 │ 0.00000000000000000000 │ 48 │

├──────┼────────────────────────┼────────────────────────┼────┤

│-0.80 │ 0.18232155927748128210 │ 0.18232155927748147639 │ 37 │

├──────┼────────────────────────┼────────────────────────┼────┤

│-0.60 │ 0.33647224087868760067 │ 0.33647224087868754516 │ 28 │

├──────┼────────────────────────┼────────────────────────┼────┤

│-0.40 │ 0.47000363483367096906 │ 0.47000363483367096906 │ 21 │

├──────┼────────────────────────┼────────────────────────┼────┤

│-0.20 │ 0.58778667152485730707 │ 0.58778667152485730707 │ 15 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 0.00 │ 0.69314718801052588315 │ 0.69314718801052588315 │ 2 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 0.20 │ 0.78845736849217618047 │ 0.78845736849217629150 │ 15 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 0.40 │ 0.87546874604624391747 │ 0.87546874604624391747 │ 21 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 0.60 │ 0.95551145419738148235 │ 0.95551145419738170439 │ 28 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 0.80 │ 1.02961942676047613077 │ 1.02961942676047613077 │ 37 │

├──────┼────────────────────────┼────────────────────────┼────┤

│ 1.00 │ 1.09861228866810956006 │ 1.09861228866810978211 │ 48 │

└──────┴────────────────────────┴────────────────────────┴────┘

1. **Дневник отладки** должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | дом | 7.12.21 | 14:00 | Забыл указать \n в строке из-за чего вся таблица съехала  printf("┌──────┬────────────────────────┬────────────────────────┬────┐\n"); | Указал \n |  |

1. **Замечания автора по существу работы:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Выводы**:

В результате выполнения данной курсовой работы я научился составлять программы на Си, которые выводят таблицу значений элементарной функции, а также вычислять функцию по формуле Тейлора

Недочеты при выполнении задания могут быть устранены следующим образом:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_