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 4. Процедуры и функции в качестве параметров

Студент: Шелаев С. И.

Группа: М8О-108Б-22,

№ по списку 24

Руководитель: Сахарин Н.А.

Дата: 09.01.23

Оценка:

Москва, 2023

**Содержание**

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

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

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

ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ …………………………………….4

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

ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ ……………………………… 4

ОПИСАНИЕ ПЕРЕМЕННЫХ, КОНСТАНТ И ПОДПРОГРАММ ……….4

ПРОТОКОЛ ………………………………………………………………….. 4

ВХОДНЫЙ ДАННЫЕ ……………………………………………………..... 5

ВЫХОДНЫЕ ДАННЫЕ ………………………………………………...…... 5

ВЫВОД ...……………………………………………………………………...5

**1. Задача**

Составить программу на языке Си с процедурами решения трансцендентных алгебраических уравнений различными способами(итераций, Ньютона и половинного деления - дихотомии). Нелинейные уравнения оформить как параметры-функции, разрешив относительно неизвестной величины в случае необходимости. Применить каждую процедуру к решению двух уравнений, заданных двумя строками таблицы, начиная с варианта с заданным номером. Если метод неприменим, дать математическое обоснование и графическую иллюстрацию.

**2. Вариант**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Уравнение | Отрезок, содержащий корень | Базовый метод | Приближенное значение корня |
| 24 |  | [1, 2] | дихотомии | 1.8756 |

**3. Общий метод решения**

Вычисление приближенного значений функций при помощи метода дихотомии, метода итераций и метода Ньютона.

Метод дихотомии - деление отрезка пополам, учитываем что знак функции должен быть разным. До тех пор, пока длина отрезка не будет меньше значения машинного эпсилон, процесс деления будет выполняться. Приближенное значение корня к моменту окончания итерационного процесса будет находиться примерно в середине заданного отрезка.

**4. Общие сведения о программе**

Аппаратное обеспечение: домашний ноутбук

Операционная система: Linux Ubuntu, версия 22.04.1 LTS

Язык и система программирования: С, GNU

**5. Функциональное назначение**

Программа предназначена для вычисления приближенного значения трансцендентных алгебраических уравнений с использованием различных численных методов и при помощи встроенных программных функций библиотеки языка Си.

**6. Описание логической структуры**

Программа получает на вход заданный отрезок, находит значение уравнения F(x) = 0 заданным методом и выводит полученный корень уравнения.

**7. Описание переменных, констант и подпрограмм**

Таблица 1. Описание функций программы

|  |  |
| --- | --- |
| Функция | Описание |
| dichotomy | Функция вычисляющая корень по методу дихотомии |
| eps | Вычисляет машинное эпсилон |
| F | Функция поступающая на вход |

Таблица 2. Описание переменных

|  |  |
| --- | --- |
| Переменная | Значение |
| double eps | Машинный эпсилон |
| double a, b | Границы отрезка |
| long double x | Значение аргумента функции |

**8. Протокол**

#include <stdio.h>

#include <math.h>

double epsilon() {

double eps = 1.0;

while (1 + eps / 2.0 != 1)

eps /= 2.0;

return eps;

}

double f\_1 (double x) {

return cos(2 / x) - 2 \* sin(1 / x) + 1 / x;

}

double f\_2 (double x) {

return sqrt(1 - (0.4 \* x \* x)) - asin(x);

}

double f\_2\_derivative(double x) {

return -((2 \* x)/(5 \* sqrt(1 - 0.4 \* x \* x))) - (1 / sqrt(2 - x \* x));

}

double dichotomy(double (\*f)(double), double a, double b, double relative\_eps, double eps) {

double x = (a + b) / 2;

if (f(a) \* f(b) < 0) {

while (fabs(a - b) > fmax(relative\_eps \* fmax(fabs(a), fabs(b)), eps)) {

x = (a + b) / 2;

if (f(x) \* f(a) < 0) {

b = x;

}

else {

a = x;

}

}

return x;

}

else {

return NAN;

}

}

double iterations(double (\*f)(double), double (\*f\_derivative)(double), double a, double b, double relative\_eps, double eps) {

double x = (a + b) / 2;

if (fabs(f\_derivative(x)) < 1)

{

while (fabs(f(x) - x) >= fmax(relative\_eps \* fmax (fabs(f(x)), fabs(x)), eps)) {

x = f(x);

}

return x;

}

else {

return NAN;

}

}

int main() {

double eps = epsilon();

double relative\_eps = sqrt(eps);

if (dichotomy(f\_1, 1, 2, relative\_eps, eps) != NAN) {

printf("%10f\n", dichotomy(f\_1, 1, 2, relative\_eps, eps));

}

if (iterations(f\_2, f\_2\_derivative, 0, 1, relative\_eps, eps) != NAN) {

printf("%10f\n", iterations(f\_2, f\_2\_derivative, 0, 1, relative\_eps, eps));

}

return 0;

}

**9. Входные данные**

Отсутствуют

**10. Выходные данные**

**serafim@serafim-VirtualBox:~/programs$ gcc -Wall test.c -g -lm -o a.out,**

**serafim@serafim-VirtualBox:~/programs$ ./a.out < test.txt**

**1.8756174**

**serafim@serafim-VirtualBox:~/programs$**

**11. Вывод**

Были изучены некоторые методы решения уравнений вида F(x) = 0. Было изучено практическое применение методов Ньютона, дихотомии и итераций для нахождения приблеженого корня уравнения.