

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский Авиационный Институт»
(Национальный Исследовательский Университет)

Институт: №8 «Компьютерные науки и прикладная математика»
Кафедра: 806 «Вычислительная математика и программирование»

Курсовая работа
по курсу «Фундаментальная
информатика»
I семестр
Задание 4
«Процедуры и функции в качестве параметров»

Группа	М8О-109Б-22
Студент	Юсуфов Р.Г.
Преподаватель	Сысоев М. А.
Оценка	
Дата	29 декабря 2022 г.

Москва, 2022

Задание

Составить программу на Си с процедурами решения трансцендентных алгебраических уравнений различными численными методами (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, например, с использованием gnuplot.

Варианты 4,5:

4	$3x - 14 + e^x - e^{-x} = 0$	[1, 3]	Ньютона	2.0692
5	$\sqrt{1-x} - \operatorname{tg} x = 0$	[0, 1]	дихотомии	0.5768

Теоретическая часть

1. Метод итераций

Идея заключается в замене исходного уравнения $f(x) = 0$ на уравнение $x = f(x)$. Перед началом итерационного процесса необходимо проверить условие сходимости: $|f'(x)| < 1$, $x \in [a, b]$. Изначально x равен $(a + b)/2$. Итерационный процесс: $x_{i+1} = f(x_i)$.

2. Метод Ньютона

Является частным случаем метода итераций, отличается условие выхода из цикла: $|f(x) * f'(x)| < (f'(x))^2$ и итерационным процессом: $x_{i+1} = x_i - (f(x_i)/f'(x_i))$

Алгоритм программы и план работы

В функции main для метода Ньютона производится проверка на сходимость. Для обоих методов нахождение значения корня рассчитывается согласно информации из условия работы.

Программа выведет значение корня.

В программе используется библиотека math.h. Для корректной работы на компиляторе gcc необходимо написать команду -lm.

В программе использовал переменные, представленные в таблице

Название переменной	Тип переменной	Смысл переменной

a	long double	Начало отрезка
b	long double	Конец отрезка
x	long double	Значения в промежутке [a;b], для которого вычисляются значения
LDBL_EPSILON	long double	Машинный эпсилон. Для long double $\varepsilon = 1.08 * 10^{-19}$

Распечатка кода программы:

Для дихотомии:

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <float.h>
```

```
long double f(long double x) {
    return sqrtl(1 - x) - tanl(x);
}
```

```
int main() {
    long double a, b;
    a = 0.01;
    b = 1.01;
    int f = 1;
    while (llabs(a-b) > LDBL_EPSILON) {
        if (f(a) * f((a + b) / 2) == 0) {
            if (f(a) == 0){
                printf("%.20Lf\n", a);
                f = 0;
                break;
            } else {
                printf("%.20Lf\n", (a + b) / 2);
                f = 0;
                break;
            }
        }
        else if (f(a) * f((a + b) / 2) > 0) {
            a = (a + b) / 2;
        } else {
            b = (a + b) / 2;
        }
    }
}
```

```

    if (f) {
        printf(" %.20Lf\n", (a + b) / 2);
    }
}

```

Для метода Ньютона:

```

#include <stdio.h>
#include <math.h>
#include <float.h>
#include <stdlib.h>

long double f(long double x) {
    return 3*x - 14 + expl(x) - expl(-x);
}

long double fproiz(long double x) {
    return 3 + expl(x) - expl(-x);
}

long double fprpr(long double x) {
    return expl(x) - expl(-x);
}

int main(void) {
    long double a = 1.0, b = 3.0, n;
    long double x = (a+b) / 2;
    if (llabs(f(x)*fprpr(x))<powl((fproiz(x)),2)) {
        while (1) {
            n = x - f(x) / fproiz(x);
            if (fabs(n - x) < LDBL_EPSILON) {
                break;
            }
            x = n;
        }
    }
    else {
        printf("Проверка на сходимость провалена\n");
    }
    printf("%.20Lf\n", x);
    return 0;
}

```

Тестирование программы

Тест №1

0.57676980757075159922

2.069218

Корни совпали. Программа работает корректно

Заключение

Были написаны функции проверки использования определенных методов

решения уравнений, определены и инициализированы сами функции различных методов поиска корня уравнения.

Данная работа полезна в увеличении познаний о способах решений уравнений и методах их программной реализации.