

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

Институт: №8 «Информационные технологии 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	Любарский И.В.
Преподаватель	Сысоев М.А.
Оценка	
Дата	

Москва, 2022

Введение

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

Задачи:

- Создать функцию, вычисляющую корень уравнения методом дихотомии.
- Создать функцию, вычисляющую корень уравнения методом итераций.
- Настроить точность вычислений.
- Считать начало и конец отрезков.
- Скомпоновать готовые функции и данные в программу вывода.

2	$\cos x - e^{-\frac{x^2}{2}} + x - 1 = 0$	[1, 2]	дихотомии	1.0804
3	$1 - x + \sin x - \ln(1 + x) = 0$	[1, 1.5]	итераций	1.1474

Дополнительная информация

Метод итераций:

Идея метода заключается в замене исходного уравнения $F(x) = 0$ на $x = f(x)$.

Достаточное условие сходимости метода: $|f'(x)| < 1, x \in [a, b]$. Это условие необходимо проверить перед началом решения задачи, так как функция $f(x)$ может быть выбрана неоднозначно, причем в случае неверного выбора указанной функции метод расходиться.

Начальное приближение корня: $x^{(0)} = (a+b)/2$ (середина исходного отрезка).

Итерационный процесс: $x^{(k+1)} = f(x^{(k)})$.

Условие окончания: $|x^{(k)} - x^{(k-1)}| < \xi$.

Приближенное значение корня: $x \approx x^{(\text{конечное})}$.

Метод дихотомии:

Очевидно, что если на отрезке $[a, b]$ существует корень уравнения, то значения функции на концах отрезка имеют разные знаки: $F(a) \cdot F(b) < 0$. Метод заключается в делении отрезка пополам и его сужении в два раза на каждом шаге итерационного процесса в зависимости от знака функции в середине отрезка.

Итерационный процесс строится следующим образом: за начальное приближение принимаются границы исходного отрезка $a^{(0)} = a, b^{(0)} = b$.

Далее вычисления проводятся по формулам: $a^{(k+1)} = (a^{(k)} + b^{(k)})/2$,
 $b^{(k+1)} = b^{(k)}$, если $F(a^{(k)}) \cdot F((a^{(k)} + b^{(k)})/2) > 0$; или по формулам: $a^{(k+1)} = a^{(k)}$,
 $b^{(k+1)} = (a^{(k)} + b^{(k)})/2$, если $F(b^{(k)}) \cdot F((a^{(k)} + b^{(k)})/2) > 0$.

Процесс повторяется до тех пор, пока не будет выполнено условие

$$|a^{(k)} - b^{(k)}| < \xi.$$

Приближенное значение корня к моменту окончания итерационного процесса получается следующим образом $x \approx (a^{(\text{конечное})} + b^{(\text{конечное})})/2$.

Практическая часть

Для решения поставленных задач необходимо инициализировать описанные функции:

- Функция варианта 2.
- Функция варианта 3.
- Функция производной.
- Функция проверка на сходимость итерационного метода.
- Функция итерационного метода.
- Функция метода дихотомии.

Также нужно определить начало и конец отрезка и учесть их при написании программы.

Название переменной	Тип переменной	Значение переменной
delta	const double	Переменная машинного нуля
a	long double	Переменная начала отрезка
b	long double	Переменная конца отрезка
Fx2	long double	Переменная аргумента функции варианта 2
Fx3	long double	Переменная аргумента функции варианта 3
Absx	long double	Переменная модуля
Ia	long double	Начала отрезка в методе итераций
Ib	long double	Конец отрезка в методе итераций
Da	long double	Начало отрезка в методе дихотомии
Db	long double	Конец отрезка в методе дихотомии
Nf	int	Номер функции

		производной
Derx	long double	Аргумент функции производной

Для уравнения варианта 3 недействительна проверка на сходимость. В точке $x \approx 1.0656$ значений модуля производной функции становится больше 1. Следовательно метод итерации неприменим в данном случае.

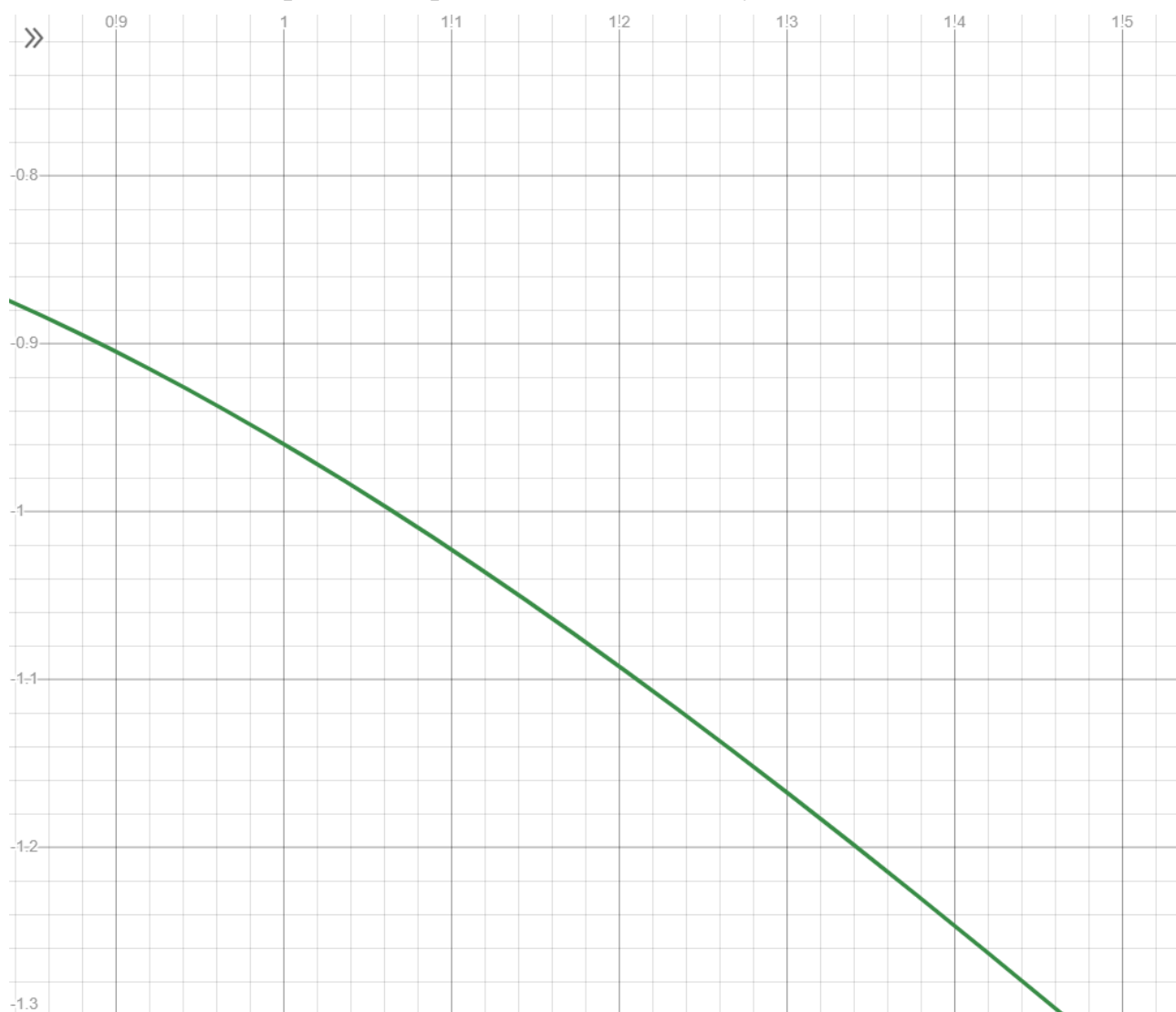


График производной функции

Алгоритм выполнения:

1. Считывание начала и конца отрезка для метода итераций.
2. Определение сходимости функции для итерационного метода.
3. Вычисление корня функции методом итераций.
4. Считывание начала и конца отрезка для метода дихотомии.
5. Вычисление корня функции методом дихотомии.
6. Вывод корней уравнений и их значения в данных точках.

Исходный код

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

long double Function2(long double x); //Возвращает значение функции 2
long double Function3(long double x); //Возвращает значение функции 3

long double derivative(int Nf, long double x); //Возвращает значение производной функции 2 или 3
long double LDabs(long double x); //Модуль для типа long double

int verify_iteration(); //Проверка на сходимость итерационного метода
long double iteration(long double a, long double b); //Решение итерационным методом

long double dihotomia(long double a, long double b); //Решение методом дихотомии

const double delta = 1.0e-15; //"Бесконечно малая" величина для машинного нуля

int main() {

    long double a, b;

    printf("Enter a b for iteration: ");
    scanf_s("%lf%lf", &a, &b);

    if (verify_iteration()) {
        printf("%lf\n", iteration(a, b));
    }
    else { printf("Function3 is not converge\n"); }

    printf("Enter a b for dihotomia: ");
    scanf_s("%lf%lf", &a, &b);
```

```

printf("Function2 x is equal : %lf\nItself equal : %lf\n", dihotomia(a, b), Function2(dihotomia(1,2)));

return 0;
}

long double Function2(long double Fx2) { return cos(Fx2) - pow(exp(1), -pow(Fx2, 2) / 2) + Fx2 - 1; }
long double Function3(long double Fx3) { return 1 - Fx3 + sin(Fx3) - log(1 + Fx3); }

long double derivative(int Nf, long double x) {
    if (Nf == 2) { return (Function2(x + delta) - Function2(x)) / delta; }
    if (Nf == 3) { return (Function3(x + delta) - Function3(x)) / delta; }
}

long double LDabs(long double Absx) {
    if (Absx < 0) { return -Absx; }
    return Absx;
}

int verify_iteration() {
    for (long double i = 1; i <= 1.5; i += delta) { if (LDabs(derivative(3, i)) >= 1) { return 0; } }
    return 1;
}

long double iteration(long double la, long double lb) {
    long double x = (la + lb) / 2;
    long double temp = Function3(x);
    while (LDabs(temp - x) >= delta) {
        x = temp;
        temp = Function3(x);
    }
    return x;
}

long double dihotomia(long double Da, long double Db) {
    while (1) {
        if (Function2(Da) * Function2((Da + Db) / 2) > 0) { Da = (Da + Db) / 2; }
        else { Db = (Da + Db) / 2; }
        if (Db - Da < delta) { return (Da + Db) / 2; }
    }
}

```

Входные данные: числа a , b начала и конца отрезка соответственно для каждого уравнения.

Выходные данные: корень уравнения и значения функции в данной точке

Протокол исполнения программы

Тест I

Ввод: 1 1.5 1 2

Вывод:

Enter a b for iteration: 1 1.5

Function3 is not converge

Enter a b for dihotomia: 1 2

Function2 x is equal : 1.089443

Itself equal : 0.000000

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.

Источники

- Метод итерации - https://ru.wikipedia.org/wiki/Метод_итерации
- Дихотомия - <https://ru.wikipedia.org/wiki/Дихотомия>