

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	Серый Н.О.
Преподаватель	Сысоев М.А.
Оценка	
Дата	

СОДЕРЖАНИЕ

1. Постановка задачи.....	3
2. Теоретическая часть.....	4
2.1. Метод Ньютона.....	4
2.2. Метод дихотомии (половинного деления).....	4
3. Практическая часть.....	
3.1. Описание алгоритма.....	
3.2. Используемые переменные.....	5
3.3. Исходный код программы.....	6
3.4. Входные и выходные данные.....	
3.5. Протокол с тестами.....	
4. Вывод.....	
5. Список используемых источников.....	

Постановка задачи

Составить программу на языке Си с процедурами решения трансцендентных алгебраических уравнений различными методами (итераций, Ньютона, дихотомии). Нелинейные уравнения оформить как параметры-функции, разрешив относительно неизвестной величины при 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.

Вариант 1:

Функция:

$$e^x + \ln x - 10x = 0$$

Отрезок, содержащий корень: [3,4]

Приближенное значение корня: 3.5265

Решение методом Ньютона.

Вариант 2:

Функция:

$$\cos x - e^{-\frac{x^2}{2}} + x - 1 = 0$$

Отрезок, содержащий корень: [1,2]

Приближенное значение корня: 1.0804

Решение методом дихотомии (половинного деления).

Теоретическая часть

Метод Ньютона

Метод Ньютона является частным случаем метода итераций.

Условие сходимости метода: $|F(x) \cdot F''(x)| < (F'(x))^2$ на отрезке $[a, b]$.

Итерационный процесс: $x^{(k+1)} = x^{(k)} - F(x^{(k)}) / F'(x^{(k)})$.

Метод дихотомии (половинного деления)

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

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

Практическая часть

Ход решения

1) Объявим в программе функции наших алгебраических уравнений и их производных.

Значения производных нужны, чтобы решить уравнение методом Ньютона.

2) Перенесём идею решения уравнений в отдельные функции Newton и Dichotomy в программу.

3) Для дополнительной проверки правильности решения выполним вычисления для вариантов обоими методами.

Используемые переменные

Название переменной	Тип переменной	Смысл переменной
x	double	Аргумент для вычисления зн. функций
a	double	Левая граница отрезка [a, b]
b	double	Правая граница отрезка [a, b]
c	double	Точка-середина отрезка [a, b]
DBL_EPSILON	double	Значение машинного эпсилон

Используемые функции

Название функции	Тип переменной	Смысл функции
func1	double	Возвращает функцию 1
func2	double	Возвращает функцию 2
derivative_func1	double	Возвращает производную 1 функции
derivative_func2	double	Возвращает производную 2 функции
f(double x)	double	Подфункция для подстановки в неё функции любого варианта на этапе вывода таблицы
d_f(double x)	double	Подфункция для подстановки в неё производной функции любого варианта на этапе вывода таблицы
Newton	double	Поиск значения функции методом Ньютона
Dichotomy	double	Поиск значения функции методом дихотомии

Исходный код программы

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

double func1(double x) {
    return exp(x) + log(x) - 10 * x;
}

double derivative_func1(double x) {
    return exp(x) + 1 / x - 10;
}

double func2(double x) {
    return cos(x) - exp(-pow(x, 2) / 2) + x - 1;
}

double derivative_func2(double x) {
    return -sin(x) + x * exp(-pow(x, 2) / 2) + 1;
}

double Newton(double f(double x1), double d_f(double x2), double a, double b) {
    double x = b;
    while (fabs(f(x) / d_f(x)) >= DBL_EPSILON) {
        x -= f(x) / d_f(x);
    }
    return x;
}

double Dichotomy(double f(double x), double a, double b) {
    double c = 0;
    while (f(c) != 0 && fabs(b - a) > DBL_EPSILON) {
        c = (a + b) / 2;
        (f(c) * f(a) > 0) ? (a = c) : (b = c);
    }
    return c;
}

int main() {
    printf("*-----*\n");
    printf("*-----Variant(#1)-----*\n");
    printf("*~::~::~::~::~::~::~::~::~::~::~::~::~::~::~*\n");
    printf("    exp(x) + log(x) - 10 * x  \n");
    printf("*-----*\n");
    printf("Newton Method value: %f\n", Newton(func1, derivative_func1, 3, 4));
    printf("Dichotomy Method value: %f\n", Dichotomy(func1, 3, 4));
    printf("*-----*\n");
    printf("*-----Variant(#2)-----*\n");
    printf("*~::~::~::~::~::~::~::~::~::~::~::~::~::~::~*\n");
    printf("cos(x) - exp(-pow(x, 2) / 2) + x - 1\n");
    printf("*~::~::~::~::~::~::~::~::~::~::~::~::~::~::~*\n");
    printf("Newton Method value: %f\n", Newton(func2, derivative_func2, 1, 2));
    printf("Dichotomy Method value: %f\n", Dichotomy(func2, 1, 2));
    printf("*-----*\n");
}
```

Входные данные

Ввод данных с клавиатуры не предусмотрен моей программой.

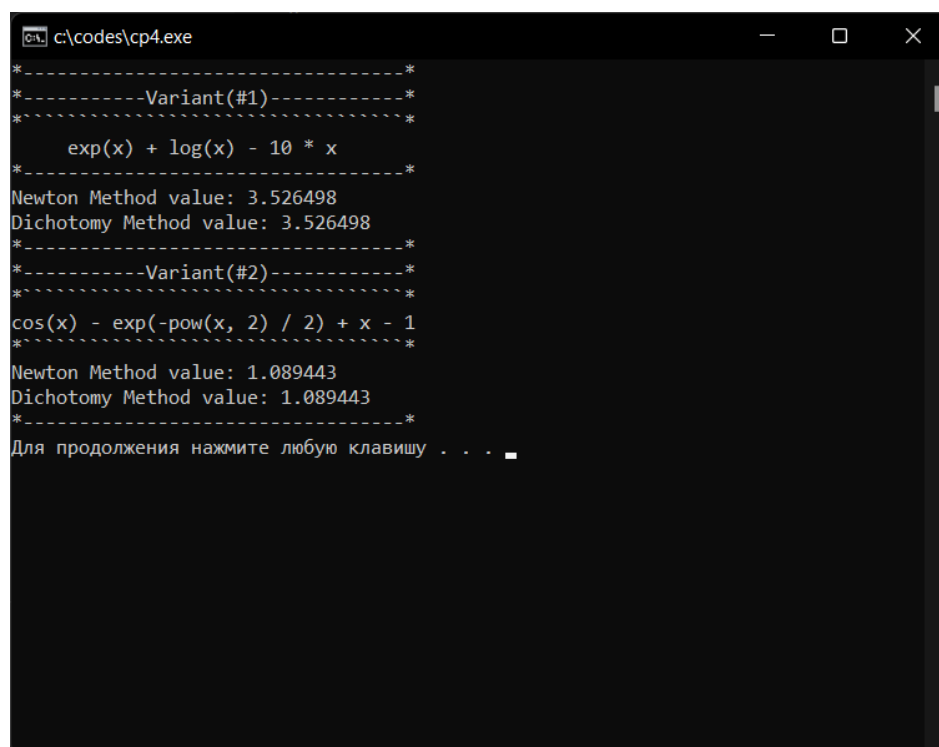
Выходные данные

Программа выводит таблицу значений для 1 варианта и 2 варианта, посчитанных методами Ньютона дихотомии.

Протокол с тестами

Тест №1

Вывод:



```
c:\codes\cp4.exe
*-----*
*-----Variant(#1)-----*
*-----*
*      exp(x) + log(x) - 10 * x
*-----*
Newton Method value: 3.526498
Dichotomy Method value: 3.526498
*-----*
*-----Variant(#2)-----*
*-----*
cos(x) - exp(-pow(x, 2) / 2) + x - 1
*-----*
Newton Method value: 1.089443
Dichotomy Method value: 1.089443
*-----*
Для продолжения нажмите любую клавишу . . .
```

Вывод

В работе описаны идеи и принципы трёх численных методов: дихотомии, итераций и Ньютона, а для методов дихотомии и Ньютона приведены и примеры решения уравнений.

Для этого была составлена программа на языке C, выводящая таблицу искомых значений.

Эlegantное решение было достигнуто и благодаря повторению правил передачи имени функции в программу.

Список используемых источников

[1] Передача имени функции в подпрограмму — URL:
<http://victor192007.narod.ru/files/cpp21.html>