

Міністерство освіти і науки України
Національний університет "Львівська політехніка"
Інститут комп'ютерних наук та інформаційних технологій
Кафедра програмного забезпечення



Звіт

Про виконання лабораторної роботи №2

На тему:

«Розв'язування нелінійних рівнянь
методом дотичних та послідовних наближень»
з дисципліни «Чисельні методи»

Лектор:

доцент каф. ПЗ
Мельник Н. Б.

Виконав:

ст. гр. ПЗ-11
Морозов О. Р.

Прийняла:

доцент каф. ПЗ
Мельник Н. Б.

« __ » _____ 2022 р.

Σ = _____ .

Львів – 2022

Тема: розв'язування нелінійних рівнянь методом дотичних та послідовних наближень

Мета: ознайомлення на практиці з методом дотичних та методом послідовних наближень для розв'язування нелінійних рівнянь.

Теоретичні відомості

Локалізація коренів - визначення інтервалів функції, на яких існує єдиний корінь.

Метод Ньютона (дотичних) - основою методу є заміна дуги кривої функції на дотичну до цієї кривої, а наближене значення кореня визначають як точку перетину дотичної з віссю абсцис.

Розглянемо метод на загальному прикладі:

1) вибрати відрізок $[a, b]$, на якому виконується умова $f(a) \cdot f''(a) > 0$ (початкова точка буде a), або $f(b) \cdot f''(b) > 0$ (початкова точка b) та $|a-b| > \varepsilon$, де ε -задана похибка;

2) обчислити $x' = x - \frac{f(x)}{f'(x)}$;

3) якщо $|x' - x| \leq \varepsilon$, то ітераційний процес завершуємо, в іншому випадку повертаємось до 2 пункту і продовжуємо ітераційний процес.

Метод простої ітерації (метод послідовних наближень) -

основою методу заміна є заміна рівняння $f(x) = 0$, $\varphi(x_0) = x_1$.

Розглянемо алгоритм методу:

1) Вибираємо довільне x , з проміжку $[a, b]$.

2) $x_{i+1} = \varphi(x_i)$, де i - ітерація.

3) Якщо $|x_{i+1} - x_i| < \varepsilon$, де i - ітерація, то пошук кореня закінчився.

В іншому випадку повертаємось до пункту 2, поки не знайдемо розв'язок із заданою точністю.

Індивідуальне завдання

Варіант 2

Написати програму розв'язку нелінійного рівняння $x^3 - 6x - 8 = 0$ методами Ньютона та послідовних ітерацій.

Хід роботи

Локалізація кореня:

Теорема 1. Якщо функція $f(x)$ є неперервною на відрізку $[a; b]$ і приймає

на його кінцях значення різних знаків, тобто виконується умова $f(a) \cdot f(b) < 0$,
то на цьому відрізку існує хоча б один корінь рівняння $f(x) = 0$.

Функція $f(x) = x^3 - 6x - 8$, та її похідна $f'(x) = 3x^2 - 6$ визначені на неперервному інтервалі на всій осі $(-\infty; +\infty)$. Визначимо інтервали монотонності. Для цього розв'яжемо рівняння $3x^2 - 6 = 0$, отже $x_1 = \sqrt{2}$; $x_2 = -\sqrt{2}$;
отже інтервали $(-\infty; -\sqrt{2})$, $(-\sqrt{2}; \sqrt{2})$, $(\sqrt{2}; +\infty)$;

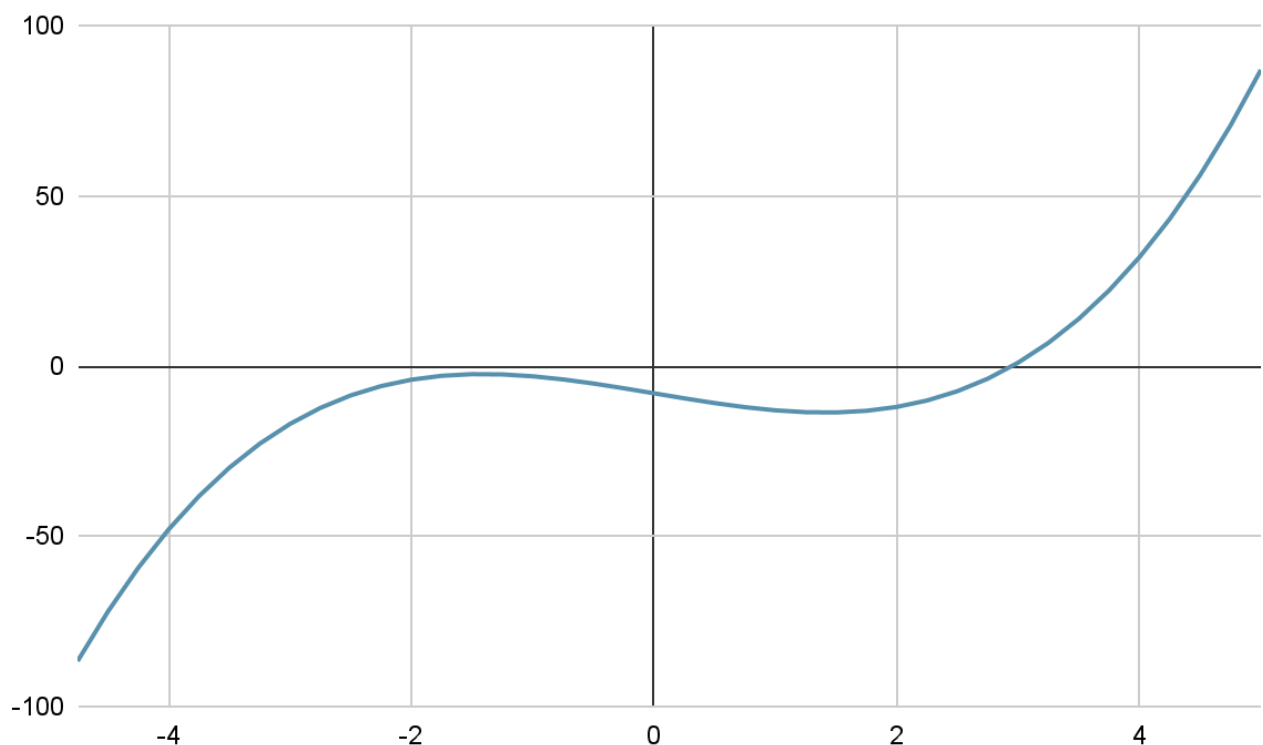
Перевіримо знак: $f(-\sqrt{2}) = -2,34$; $f(\sqrt{2}) = -13,66$;

Отже знак змінюється тільки на проміжку $(\sqrt{2}; +\infty)$;

Перевіримо згідно Теорема 1 $f(\sqrt{2}) \cdot f(4) < 0$;

Отже згідно теорема 1 єдиний корінь є на проміжку $(\sqrt{2}; +\infty)$.

Розв'язок графічним способом:



Код програми

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <math.h>

double f(double x);
double fp(double x);
double fpp(double x);
double fi(double x, double a, double b);

void retryInput();

void NewtonMethod(double n, double m, double eps);
void IterationMethod(double a, double b, double eps);

int main() {

    std::cout << "Here we have an Expression:  $x^3 - 6x - 8 = 0$ \n" <<
    "Now you have to enter a limits and eps\n" << "Expression has 1 root
    between (2,5; 3)\n";

    double a = 0, b = 0, eps = 0;
    while (1){
        printf("Enter Eps:");
        while (scanf("%lf", &eps) != 1 && eps <= 0){
            retryInput();
            printf("Wrong input\ntry again:");
        }
        printf("Enter A:");
        while (scanf("%lf", &a) != 1){
            retryInput();
            printf("Wrong input\nTry again:");
        }
        printf("Enter B:");
        while (scanf("%lf", &b) != 1){
            retryInput();
            printf("Wrong input\nTry again:");
        }
        if ((f(a) * f(b)) >= 0 || fabs(a - b) < eps){
            printf("Limits not correct\nTry again\n");
        }
        else{
            break;
        }
    }
}
```

```

        if (f(a) * fpp(a) > 0) {
            NewtonMethod(a, b, eps);
        }
        else if(f(b) * fpp(b) > 0) {
            NewtonMethod(b, a, eps);
        }
        IterationMethod(a, b, eps);
        return 0;
    }

double f(double x) {
    return ((x * x * x) - (6 * x) - 8);
}

double fp(double x) {
    return ((3 * x * x) - 6);
}

double fpp(double x) {
    return 6 * x;
}

void retryInput(){
    while (getchar() != '\n');
}

void NewtonMethod(double n, double m, double eps) {
    double x = n;
    double xi = x;
    x = xi - (f(xi)) / (fp(xi));
    int i = 1;
    while (fabs(xi - x) > eps) {
        xi = x;
        x = xi - (f(xi)) / (fp(xi));
        i++;
    }
    printf("\tNewton Method\n\tSolution\n\n X = \t%.10lf\nf(x)
= \t%.10lf \nIteration = %d\n", x, f(x), i);
}

void IterationMethod(double a, double b, double eps) {
    double x = b, xi = fi(x, a, b);
    int i = 1;
    while (fabs(xi - x) >= eps) {
        x = xi;
        xi = fi(x, a, b);
        i++;
    }
}

```

```

    }
    printf("\tNewton Method\n\tSolution\n\n X = \t%.10lf\nf(x)
=\t%.10lf \nIteration = %d\n", x, f(x), i);
}

double fi(double x, double a, double b) {

    double k;
    if ( a >= 0 && b >= 0) {
        k = fp(b) / (4./3);
    }
    else if(a <= 0 && b <= 0) {
        k = -1 * fp(b) / (4./3);
    }
    else {
        if (a <= 0) {
            a = 0;
        }
        else {
            b = 0;
        }
        k = fp(b) / 6;
    }
    return (x - (f(x) / k));
}

```

Результат работы

```

Консоль отладки Microsoft Visual Studio
Here we have an Expression: x^3 - 6x - 8 = 0
Now you have to enter a limits and eps
Expression has 1 root between (2,5; 3)
Enter Eps:0.001
Enter A:0
Enter B:6
Newton Method
Solution

X = 2.9513730356
f(x) = 0.0000000000
Iteration = 6
Newton Method
Solution

X = 2.9517142134
f(x) = 0.0068695573
Iteration = 26

D:\University\2_semester\Чисельн? методи\Lab 2\64\Debug\Lab 2.exe (процесс 9288) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

```

Висновок

Виконуючи лабораторну роботу №2, я навчився програмувати пошук наближеного розв'язку нелінійних рівнянь методом Ньютона (дотичних) та методом простої ітерації з вказаною точністю.