****

**MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII**

**AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Informatică şi Ingineria Sistemelor**

**Calancea Catalin**

**MI-222**

**Raport**

**pentru lucrarea de laborator Nr.1**

***la cursul de “Metode Numerice”***

Verificat:

lector universitar

Buzurniuc Stefan

**Chișinău – 20****23**

**CUPRINS**

[1.INTRODUCERE 3](#_Toc149642608)

[2.SARCINA 4](#_Toc149642615)

[3.COD 4](#_Toc149642616)

[4.CONCLUZIA 6](#_Toc149642617)

# 1.INTRODUCERE

# Metodele numerice au devenit fundamentale în rezolvarea unei game variate de probleme matematice și științifice, inclusiv în domenii precum inginerie, fizică, economie și multe altele. Aceste metode reprezintă un instrument esențial pentru găsirea soluțiilor numerice la ecuații complexe sau sisteme de ecuații, care deseori nu pot fi rezolvate analitic. În această lucrare de laborator, ne propunem să explorăm metodele numerice utilizate pentru găsirea rădăcinilor reale ale ecuațiilor algebrice și să analizăm performanța acestor metode în diverse contexte.

# Scopul acestei lucrări este de a înțelege și de a aplica metodele numerice pentru determinarea rădăcinilor reale ale unei funcții, în contextul unei funcții reale de variabilă reală, y = f(x). Mai precis, lucrarea urmărește următoarele obiective:

# 1) Separarea tuturor rădăcinilor reale ale ecuației f(x) = 0, pentru a identifica intervalele în care acestea se găsesc.

# 2) Determinarea unei rădăcini reale a ecuației cu o eroare mai mică decât ε = 10^(-2), utilizând metoda înjumătățirii intervalului.

# 3) Precizarea rădăcinii obținute cu o exactitate ε = 10^(-6), prin aplicarea a trei metode distincte: metoda aproximărilor succesive, metoda tangentelor (Newton) și metoda secantelor. Această parte a lucrării va analiza și compara numărul de iterații necesare și evaluările funcției și a derivatelor pentru fiecare metodă.

# Această lucrare de laborator oferă oportunitatea de a explora aplicarea practică a metodelor numerice în rezolvarea ecuațiilor algebrice și de a înțelege avantajele și limitările fiecărei metode. Rezultatele obținute vor oferi o perspectivă asupra eficienței și precisiei diferitelor tehnici utilizate în analiza numerică, cu impact în rezolvarea problemelor reale din domeniile științei și ingineriei.

# 2.SARCINA

Se considera ecuatia neliniara f(x) = 0.

1. Sa se separe radacinile ecuatiei din intervalul (-5, 5).
2. Sa se determine solutia cu precizia ε = 10-6 prin metodele a) bisectiei ; b) metoda Newton; c) metoda aproximatilor successive; d) metoda secantelor.

0.1ex + x3 -2x + 2 = 0

# 3.COD

#include <iostream>

#include <cmath>

#include <iomanip>

#define E 2.71828182846

using namespace std;

double f(double x) *//ecuatia*

{

return 0.1 \* pow(E, x) + pow(x,3) - 2 \* x + 2 ;}

double df(double x) *//derivata ecuatiei*

{

return (pow(E,x)/10) + 3 \* pow(x,2)-2;

}

double metodaBisectiei(double a, double b, double epsilon, int*&* iteratii)

{

double x = a;

while ((b - a) >= epsilon)

{

x = (a + b) / 2;

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

{

b = x;

}

else

{

a = x;

}

iteratii++;

}

return x;

}

double metodaNewton(double x0, double epsilon, int*&* iteratii)

{

double x = x0;

while (fabs(f(x)) >= epsilon)

{

x = x - f(x) / df(x); *//formula metodei lui Newton*

iteratii++;

}

return x;

}

double metodaAproxSuccesive(double x0, double epsilon, int*&* iteratii)

{

double x = x0;

while (fabs(f(x) - x) >= epsilon)

{

x = f(x); *//formula metodei aproximaatiilor succesive*

iteratii++;

}

return x;

}

double metodaSecantelor(double x0, double x1, double epsilon, int*&* iteratii)

{

double x = x1;

double x\_prev = x0;

while (fabs(f(x)) >= epsilon)

{

double f\_x = f(x);

double f\_x\_prev = f(x\_prev);

double x\_next = x - f\_x \* (x - x\_prev) / (f\_x - f\_x\_prev); *//formula metodei secantelor*

x\_prev = x;

x = x\_next;

iteratii++;

}

return x;

}

int main() {

double epsilon = 1e-6;

double a = -5.0, b = 5.0; *//inceputul si sfarsitul intervalului*

int numRadacini = 0; *//numarul de radacini*

cout << "Radacinile se afla pe intervalele:\n";

for (double x = a; x < b; x += 0.01)

{

double fx = f(x);

double fx1 = f(x + 0.01);

if (fx \* fx1 < 0)

{

numRadacini++;

cout << numRadacini << ". Intervalul [" << setprecision(3) << x << ", " << x + 0.01 << "]" << endl;

int iteratiiBisectie = 0;

double rezBisectie = metodaBisectiei(x, x + 0.01, epsilon, iteratiiBisectie);

cout << "\n Iteratii total metoda Bisectiei: " << iteratiiBisectie << endl;

cout << "Solutia metoda Bisectiei: " << setprecision(6) << rezBisectie << endl;

int iteratiiNewton = 0;

double rezNewton = metodaNewton((x + x + 0.01) / 2, epsilon, iteratiiNewton);

cout << "\nIteratii total metoda Newton: " << iteratiiNewton << endl;

cout << "Solutia metoda Newton: " << setprecision(6) << rezNewton << endl;

int iteratiiAproxSucc = 0;

double rezAproximSucc = metodaAproxSuccesive((x + x + 0.01) / 2, epsilon, iteratiiAproxSucc);

cout << "\nIteratii total metoda Aproximatiilor Succesive: " << iteratiiAproxSucc << endl;

cout << "Solutia metoda Aproximatiilor Succesive: " << setprecision(6) << rezNewton << endl;

int iteratiiSecante = 0;

double rezSecante = metodaSecantelor(x, x + 0.01, epsilon, iteratiiSecante);

cout << "\nIteratii total metoda Secantelor: " << iteratiiSecante << endl;

cout << "Solutia metoda Secantelor: " << setprecision(6) << rezSecante << endl;

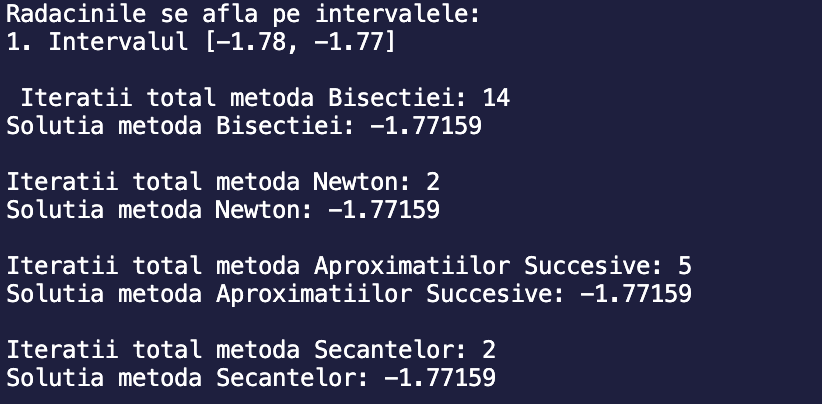
cout << "\n";

}

}

return 0;

}



# 4.CONCLUZIA

În cadrul acestei lucrări de laborator, ne-am propus să studiem și să aplicăm diferite metode pentru găsirea rădăcinilor ecuațiilor algebrice, abordând mai multe aspecte, precum separarea rădăcinilor reale, determinarea unei rădăcini cu o anumită precizie și compararea performanței metodelor utilizate.

1) Pentru separarea tuturor rădăcinilor reale ale ecuației f(x) = 0, am folosit metoda intervalului și metoda bisecției pentru a identifica intervalele în care se găsesc rădăcinile, apoi am aplicat metoda bisecției pentru a obține o estimare a rădăcinilor cu o eroare inițială.

2) Pentru a determina o rădăcină reală a ecuației cu o eroare mai mică decât ε = 10^(-2), am aplicat metoda înjumătățirii intervalului, care ne-a permis să obținem o valoare aproximativă a rădăcinii cu precizie.

3) Pentru a obține o rădăcină cu o precizie superioară, ε = 10^(-6), am aplicat trei metode de rafinare a rădăcinilor: metoda aproximațiilor succesive, metoda tangentelor (Newton) și metoda secantelor. Am observat că toate aceste metode au dus la rezultate satisfăcătoare în ceea ce privește precizia, dar au diferit în ceea ce privește numărul de iterații necesare și evaluările funcției și a derivatelor.

4) Comparând rezultatele obținute, am observat că metoda Newton a dat cele mai rapide convergențe la rădăcina exactă, dar necesită evaluări ale derivatei, ceea ce poate fi costisitor în unele situații. Metoda secantelor este o alternativă bună dacă derivata nu este ușor de obținut. Metoda aproximărilor succesive, deși poate necesita mai multe iterații, este o opțiune valabilă pentru ecuații unde celelalte metode pot fi dificil de aplicat.

În concluzie, această lucrare de laborator a arătat importanța alegerii metodei potrivite pentru găsirea rădăcinilor ecuațiilor algebrice în funcție de cerințe precum precizia, numărul de iterații și evaluările funcției și a derivatelor. Acesta oferă o bază solidă pentru înțelegerea și aplicarea metodelor numerice în rezolvarea problemelor practice care implică ecuații.