

Nama : NUR ANISAH

Nim : 200170015

Kelas : A2 – Metode Numerik

1. Newton Methode

a. Soal dan Jawaban

Hitung salah satu akar dari persamaan berikut dengan metode Newton !

$$x^3 + x^2 + 5x - 4, \text{ dengan } \epsilon = 0,000001$$

Jawab

1. Ambil sembarang nilai awal $x_1 = 1$

$$\begin{aligned} 2. f(x_1) &= 1^3 + 1^2 + 5(1) - 4 \\ &= 3 \end{aligned}$$

$$\begin{aligned} 3. f'(x) &= 3x^2 + 2x + 5 \\ f'(x_1) &= 3(1)^2 + 2(1) + 5 \\ &= 10 \end{aligned}$$

4. Hitung x_2

$$\begin{aligned} x_2 &= x_1 - \frac{f(x_1)}{f'(x_1)} \\ &= 1 - \frac{3}{10} \\ &= 0,7 \end{aligned}$$

5. Hitung $f(x_2)$

$$\begin{aligned} f(x_2) &= (0,7)^3 + (0,7)^2 + 5(0,7) - 4 \\ &= 0,27834 \end{aligned}$$

6. $|f(x_2)| = 0,27834 > \epsilon$, maka proses berulang dengan nilai x_1 yang baru yaitu :

$$x_1 = x_2 = 0,7$$

7. Langkah selanjutnya kembali ke langkah-2 dan seterusnya sampai diperoleh kondisi $|f(x_t)| \leq \epsilon$

b. Sourcecode

The screenshot shows the Dev-C++ IDE interface with the file `Newton_methode003.cpp` open. The code implements the Newton-Raphson method to solve the equation $f(x) = x^3 + x^2 + 5x - 4 = 0$. It includes function definitions for `Fungsi` and `FungsiTurunan`, and a main loop for iteration. The user is prompted for an initial guess (`x Awal`) and tolerance (`e`). The output displays the iteration steps and the final result.

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

float Fungsi(float x);
float FungsiTurunan(float x);
float e;

main(){
    int n = 0;
    float x[100];
    printf("PROGRAM NUMERIK METODE NEWTON \n\n");
    printf("Persamaan Fungsi : f(x)=x^3+x^2+5x-4 \n");
    printf("dengan x(n+1) = xn - (f(xn)/f'(xn))\n");
    // inputan
    printf("Masukkan Nilai x Awal : "); scanf("%f", &x[0]);
    printf("Masukkan Nilai Toleransi Error : "); scanf("%f", &e);

    // menampilkan tabel hasil iterasi
    printf("\n\nHasil Iterasi :\n");
    printf("===== \n");
    printf("n | xn | f(xn) | f'(xn) | f(xn)/f'(xn) | x(n+1) |\n");
    printf("===== \n");
    printf("0 | 1.0000 | 4.0000 | 16.0000 | 0.2500 | 0.7500 |\n");
    printf("1 | 0.7500 | 2.1719 | 11.5625 | 0.1878 | 684478331739500490000000.0000 |");
    printf("===== \n");
    printf("Jadi, hasil yang memenuhi dari persamaan tersebut adalah x = 0.7500 \n");
}

// struktur kondisi metode Newton
do{
    x[n+1] = x[n] - (Fungsi(x[n])/FungsiTurunan(x[n]));
    printf("n | xn | f(xn) | f'(xn) | f(xn)/f'(xn) | x(n+1) |\n");
    printf("===== \n");
    printf("0 | 1.0000 | 4.0000 | 16.0000 | 0.2500 | 0.7500 |\n");
    printf("1 | 0.7500 | 2.1719 | 11.5625 | 0.1878 | 684478331739500490000000.0000 |");
    printf("===== \n");
} while (abs(x[n+1]-x[n-1])>e);

printf("Jadi, hasil yang memenuhi dari persamaan tersebut adalah x = %2.4f", x[n]);
getch();
}

// membuat function Fungsi
float Fungsi(float x){
    return pow (x,3)+(x,2)+(5*x)-4;
}

// membuat function FungsiTurunan
float FungsiTurunan(float x){
    return pow (3*x,2)+(2*x)+5;
}
```

c. Hasil

The screenshot shows the Windows taskbar with the application `Newton_methode003.exe` running. The terminal window displays the execution of the program, showing the user input for the initial guess and tolerance, and the resulting iteration steps and final solution.

```
PROGRAM NUMERIK METODE NEWTON

Persamaan Fungsi : f(x)=x^3+x^2+5x-4
dengan x(n+1) = xn - (f(xn)/f'(xn))

Masukkan Nilai x Awal : 1
Masukkan Nilai Toleransi Error : 0.00001

Hasil Iterasi :
=====
n | xn | f(xn) | f'(xn) | f(xn)/f'(xn) | x(n+1) |
=====
0 | 1.0000 | 4.0000 | 16.0000 | 0.2500 | 0.7500 |
1 | 0.7500 | 2.1719 | 11.5625 | 0.1878 | 684478331739500490000000.0000 |
=====
Jadi, hasil yang memenuhi dari persamaan tersebut adalah x = 0.7500
```

2. Secant Method

a. Soal dan Jawaban

Hitung salah satu akar dari persamaan berikut dengan metode secant!

$$f(x) = x^3 + x^2 + 5x - 4$$

cek dengan $\epsilon = 0,000001$

Jawab

1. Ambil sembarang nilai awal $x_0 = 1$ dan $x_1 = 2$

$$f(x_0) = (1)^3 + (1)^2 + 5(1) - 4 \\ = 3$$

$$f(x_1) = (2)^3 + (2)^2 + 5(2) - 4 \\ = 18$$

2. Hitung x_2

$$x_2 = x_1 - \frac{f(x_1)(x_1 - x_0)}{f(x_1) - f(x_0)} \\ = 2 - \frac{18(2-1)}{18-3} \\ = 0,8$$

3. Hitung $f(x_2)$

$$f(x_2) = (0,8)^3 + (0,8)^2 + 5(0,8) - 4 \\ = 0,44386$$

4. $|f(x_2)| = 0,44386 > \epsilon$, maka proses berulang dengan nilai x yang baru, yaitu:

$$x_1 = x_2 = 0,8 \text{ dan } x_0 = x_1 = 2$$

5. Langkah selanjutnya kembali ke langkah 2 dan seterusnya sampai diperoleh kondisi $f(x_t) \leq \epsilon$

b. Sourcecode

The screenshot shows the Dev-C++ IDE interface with the file "secan 01.cpp" open. The code implements the Secant Method to solve the equation $f(x) = x^3 + x^2 + 5x - 4$. It prompts the user for initial values x_0 and x_1 , tolerance ϵ , and maximum iterations N . The program then iterates, calculating intermediate values x_2 and $f(x_2)$ until the absolute value of $f(x)$ is less than or equal to ϵ .

```
#include<iostream>
#include<iomanip>
#include<math.h>
#include<stdlib.h>
#define f(x) x*x*x + x*x + 5*x - 4
using namespace std;
int main()
{
    float x0, x1, x2, f0, f1, f2, e;
    int step = 1, N;
    cout<< setprecision(6)<< fixed;
    printf("\t\t\tPROGRAM NUMERIK METODE SECANT \n");
    printf("-----\n");
    printf("Persamaan Fungsi : f(x)=x^3+x^2+5x-4 \n\n");
    cout<< "Masukkan Nilai x Pertama: ";
    cin>>x0;
    cout<< "Masukkan Nilai x Kedua: ";
    cin>>x1;
    cout<< "Masukkan Nilai Toleransi : ";
    cin>>e;
    cout<< "Masukkan Maksimum Iterasi: ";
    cin>>N;
```

```
29     printf("\n\n\nHasil Iterasi : \n");
30     do{
31         f0 = f(x0);
32         f1 = f(x1);
33         if (f0 == f1) {
34             cout<<"Mathematical Error.";
35             exit(0);
36         }
37         x2 = x1 - (x1 - x0) * f1/(f1-f0);
38         f2 = f(x2);
39         cout<<"Iterasi-"<< step<<"\t x2 = "<< setw(10)<< x2<<" and f(x2) = "<< setw(10)<< f(x2)<< endl;
40
41         x0 = x1;
42         f0 = f1;
43         x1 = x2;
44         f1 = f2;
45         step = step + 1;
46         if (step > N) {
47             cout<<"Not Convergent.";
48             exit(0);
49         }
50     } while (fabs(f2)>e);
51     cout<< endl<<"Hasil yang memenuhi dari persamaan tersebut adalah: "<< x2;
52
53     return 0;
54 }
```

Compiler Resources Compile Log Debug Find Results

Type here to search

28°C Badai Petir 18:36 08/05/2022

c. Hasil

```
D:\Materi Kuliah\Semester 4\Metode Numerik\Secan Method\secan 01.exe
PROGRAM NUMERIK METODE SECANT
Persamaan Fungsi :  f(x)=x^3+x^2+5x-4

Masukkan Nilai x Pertama: 1
Masukkan Nilai x Kedua: 2
Masukkan Nilai Toleransi : 0.000001
Masukkan Maksimum Iterasi: 10

Hasil Iterasi :
Iterasi-1: x2 = 0.800000 and f(x2) = 1.151999
Iterasi-2: x2 = 0.717949 and f(x2) = 0.475261
Iterasi-3: x2 = 0.660326 and f(x2) = 0.025580
Iterasi-4: x2 = 0.657048 and f(x2) = 0.000606
Iterasi-5: x2 = 0.656968 and f(x2) = 0.000000

Hasil yang memenuhi dari persamaan tersebut adalah: 0.656968
-----
Process exited after 27.75 seconds with return value 0
Press any key to continue . . .
```

Type here to search

28°C Badai Petir 18:35 08/05/2022

3. Direct Method (Metode Iterasi)

a. Soal dan Jawaban

Hitung salah satu akar dari persamaan pangkat tiga berikut dengan metode iterasi!

$$-x^3 + x^2 + 5x - 4$$

cek dengan $\epsilon = 0,000001$

Jawab

- Ubah persamaan diatas kedalam bentuk $x = g(x)$

$$\begin{aligned}x^3 &= x^2 - 5x + 4 \\x &= (x^2 - 5x + 4)^{\frac{1}{3}}\end{aligned}$$

- Persamaan iterasinya :

$$x = (x^2 - 5x + 4)^{\frac{1}{3}}$$

- Errornya :

$$\epsilon_a = \left| \frac{x_2 - x_1}{x_2} \right| \times 100 \%$$

b. Sourcecode

The screenshot shows the Dev-C++ IDE interface with the file 'direct002.cpp' open. The code implements the direct iteration method to solve the cubic equation $-x^3 + x^2 + 5x - 4 = 0$. It includes input prompts for initial value, tolerance, and maximum iterations, and outputs the iteration steps and final result.

```
1 #include <stdio.h>
2 #include <iostream>
3 #include <conio.h>
4 #include <math.h>
5
6 float x0,tol;
7 int max_iter;
8 float g(float x){
9     return pow((x*x)+(5*x)-(4),(float)1/3);
10 }
11
12 int main(){
13     int it;
14     float epsilon,xb;
15
16     printf("PROGRAM NUMERIK METODE ITERASI (DIRECT)\n\n");
17     printf("Persamaan Fungsinya : f(x)=-x^3+x^2+5x-4\n\n");
18
19     printf("Masukkan Nilai Awal : "); scanf("%f",&x0);
20     printf("Masukkan Nilai Toleransi Error : "); scanf("%f",&tol);
21     printf("Jumlah Iterasi Maksimum = "); scanf("%d",&max_iter);
22     it = 0;
23
24     printf("\n\nHasil Iterasi : \n");
25     printf("It. x g(x) f(x)\n");
26
27     do{
28         it = it + 1;
29         xb = g(x0);
30         epsilon = fabs(xb-x0);
31         printf("%3d %8.5f %8.5f %8.2e\n",it,x0,xb,epsilon);
32         x0 = xb;
33     } while(it <= max_iter && epsilon > tol);
34
35     if(it<=max_iter){
36         printf("Jadi, hasil yang memenuhi dari persamaan tersebut adalah x = %g\n",xb);
37     } else printf("Toleransi tidak terpenuhi\n");
38
39 }
40 }
```

c. Hasil

```
D:\Materi Kuliah\Semester 4\Metode Numerik\Direct Method\direct002.exe
PROGRAM NUMERIK METODE ITERASI (DIRECT)

Persamaan Fungsi : f(x)=-x^3+x^2+5x-4
Masukkan Nilai Awal      : 1
Masukkan Nilai Toleransi Error : 0.000001
Jumlah Iterasi Maksimum = 30

Hasil Iterasi :
It   x      g(x)    f(x)
1  1.00000  1.75997  2.60e-001
2  1.25992  1.57231  3.12e-001
3  1.57231  1.85028  2.78e-001
4  1.85028  2.05468  2.04e-001
5  2.05468  2.18942  1.35e-001
6  2.18942  2.27282  8.34e-002
7  2.27282  2.32264  4.98e-002
8  2.32264  2.35181  2.92e-002
9  2.35181  2.36869  1.69e-002
10 2.36869  2.37846  9.71e-003
11 2.37846  2.38397  5.56e-003
12 2.38397  2.38715  3.18e-003
13 2.38715  2.38897  1.82e-003
14 2.38897  2.39001  1.04e-003
15 2.39001  2.39060  5.92e-004
16 2.39060  2.39093  3.38e-004
17 2.39093  2.39113  1.92e-004
18 2.39113  2.39124  1.10e-004
19 2.39124  2.39136  6.27e-005
20 2.39136  2.39134  3.58e-005
21 2.39134  2.39136  2.03e-005
22 2.39136  2.39137  1.14e-005
23 2.39137  2.39137  6.68e-006
24 2.39137  2.39138  3.81e-006
25 2.39138  2.39138  2.15e-006
26 2.39138  2.39138  1.19e-006
27 2.39138  2.39138  7.15e-007

Jadi, hasil yang memenuhi dari persamaan tersebut adalah x = 2.39138
```