

LAPORAN PRAKTIKUM MATA KULIAH
METODE NUMERIK
PRAKTIKUM 10 – INTERPOLASI POLINOMIAL



DISUSUN OLEH:

M0521003 – ADI PRASETYA

PROGRAM STUDI INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET

2022

BAB I

ANALISIS SOURCE CODE

1. ANALISIS NILAI AWAL

x (m)	y (m)
2,0000	7,2000
4,2500	7,1000
5,2500	6,0000
7,8100	5,0000
9,2000	3,5000
10,6000	5,0000

Pada soal terdapat data dari nilai x (m) dan y (m) seperti tabel di atas.

x (m)	y (m)	selisih
2,0000	7,2000	5,5000
4,2500	7,1000	3,2500
5,2500	6,0000	2,2500
7,8100	5,0000	0,3100
9,2000	3,5000	1,7000
10,6000	5,0000	3,1000

Berdasarkan hasil dari analisis nilai awal untuk pencarian $f(7.5)$ dengan menggunakan fungsi polynomial orde-2 maka akan diperlukan 3 nilai awal dalam perhitungan ini. Untuk menentukan nilai awal dilakukan dengan mencari 3 nilai awal yang memiliki selisih terkecil hingga terbesar dari **nilai x** yang dicari. Nilai awal yang memiliki selisih terkecil dengan $x = 7.5$ adalah **5.25, 7.81, dan 9.2**. Dengan demikian, ketiga nilai awal tersebut akan digunakan dalam perhitungan ini karena memiliki selisih terkecil dengan **nilai x** dibandingkan dengan nilai awal yang lainnya.

2. ANALISIS SOURCE CODE LAGRANGE

```
1 % M0521003 - Adi Prasetya
2
3 function itp=lagrange(xy)
4     x=xy(:,1);
5     y=xy(:,2);
6     itp="";
7     for i=1:length(x) %Menghitung elemen summation
8         L="";
9         for j=1:length(x) %Menghitung produk Li
10            if j==i continue endif
11            L=strcat(L,"(x-",num2str(x(j)),")./((",num2str(x(i)-x(j)),").*)");
12        endfor
13        L=strcat(L,num2str(y(i)));
14        itp=strcat(itp,"+",L);
15    endfor
16    itp=str2func(strcat("@(x)",itp(2:length(itp))));
17 end
18
```

Pada *source code* di atas terdapat *function* **lagrange(xy)** yang dimana hasil operasi dalam *function* tersebut akan di-*return* dalam variabel **lagranng** dengan parameter *xy* yang berupa matriks. Kemudian pada **line 4** dan **5** dilakukan pemisahan untuk nilai *x* dan *y* dimana nilai *x* akan dimasukkan ke dalam variabel *x* dan nilai *y* akan dimasukkan ke dalam variabel *y*. Pada **line 6** terdapat inisialisasi dari variabel *itp* dengan *null*. Lalu, pada **line 7** terdapat *for loop* yang dimulai dari *i* = 1 hingga *i* = panjang dari *x* untuk menghitung elemen *summation*. Dalam *for loop* tersebut terdapat inisialisasi dari variabel *L* dengan *null* dan *for loop* yang dimulai dari *j* = *i* hingga *i* = panjang dari *x* untuk menghitung produk dari ***L_i***. Dalam *for loop* tersebut terdapat *if condition* jika nilai dari *i* sama dengan nilai dari *j* maka akan dilakukan *continue* dan *men-skip* *source code* yang ada di bawahnya serta kembali ke *for loop* yang berada di dalam. Hal ini dikarenakan tidak boleh produk ***L_i*** yang *j* sama dengan *i*. Kemudian jika nilai *j* tidak sama dengan nilai *i* maka akan dilakukan operasi untuk pencarian nilai ***L_i*** (*x*) dengan rumus sebagai berikut.

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}$$

Kemudian, jika produk ***L_i*** sudah ditemukan dan *for loop* berhenti, maka akan dilakukan pencarian nilai ***f_i*** (*x*) dengan rumus sebagai berikut.

$$f_i(x) = \sum_{i=0}^n L_i(x) f(x_i)$$

Setelah itu, nilai dari variabel ***itp*** yang sebelumnya diinisialisasi dengan *null* dijumlahkan dengan nilai ***f_i*** (*x*) untuk menghitung *summation*, lalu dilakukan iterasi berikutnya. Setelah *for loop* untuk menghitung *summation* berhenti, selanjutnya akan dilakukan penghilangan nilai dari “+” yang tidak digunakan pada **line 14** dengan *itp(2:length(itp))*. Kemudian, dilakukan perubahan *string itp* yang memuat *string* fungsi polinomial hasil interpolasi menjadi fungsi *anonymous* dengan parameter (*x*) sebagai variabel independen. Lalu, *itp* akan di-*return* sebagai *anonym function*.

3. ANALISIS SOURCE CODE NEWTON DIVIDED-DIFFERENCE

```
1 % M0521003 - Adi Prasetya
2
3 function itp=ndd(xy)
4     x=xy(:,1);
5     y=xy(:,2);
6     for i=2:length(x) %Membuat tabel untuk NDD
7         for j=i:length(x)
8             y(j,i)=(y(j,i-1)-y(j-1,i-1))/(x(j)-x(j-(i-1)));
9         endfor
10    endfor
11    itp=num2str(y(1,1));
12    temp="";
13    for i=1:length(x)-1 %Merumuskan persamaan hasil interpolasi
14        temp=strcat(temp,".*(x-",num2str(x(i)),")");
15        itp=strcat(itp,"+",num2str(y(i+1,i+1)),temp);
16    endfor
17    itp=str2func(strcat("@(x)",itp));
18 end
19
```

Pada *source code* di atas terdapat *function* **ndd(xy)** yang dimana hasil operasi dalam *function* tersebut akan di-*return* dalam variabel **itp** dengan parameter xy yang berupa matriks. Kemudian pada **line 4** dan **5** dilakukan pemisahan untuk nilai x dan y dimana nilai x akan dimasukkan ke dalam variabel x dan nilai y akan dimasukkan ke dalam variabel y. Lalu, pada **line 6** terdapat *for loop* yang dimulai dari $i = 2$ hingga $i =$ panjang dari x. Dalam *for loop* tersebut terdapat *for loop* yang dimulai dari $j = i$ hingga $j =$ panjang dari x yang berisi rumus sebagai berikut.

$$y_{j,i} = \frac{y_{j,i-1} - y_{j-1,i-1}}{x_j - x_{j-(i-1)}}$$

Kemudian, terdapat variabel **itp** untuk menampung tabel *newton divide different* dimana nilai $y(1,1)$ akan dieksekusi dengan *function* **num2str** dan hasilnya akan disimpan dalam variabel **itp**. Nilai (1,1) karena yang digunakan adalah nilai diagonal. Lalu, terdapat *for loop* yang dimulai dari $i = 1$ hingga $i =$ Panjang x dimana di dalamnya terdapat variabel **temp** yang berisi perkalian oleh $x - x_i$ dan nilai x paling akhir tidak digunakan. Setelah dilakukan penyusunan satu per satu, maka string akan diubah menjadi sebuah *function* dengan *function* **str2func()**. Kemudian, hasilnya akan disimpan ke dalam variabel **itp**.

BAB II

ANALISIS PRAKTIKUM (COMMAND WINDOW)

1. ANALISIS PRAKTIKUM LAGRANGE

```
Command Window
>> xy = [5.25 6.0;7.81 5.0;9.2 3.5]
xy =

    5.2500    6.0000
    7.8100    5.0000
    9.2000    3.5000
```

Dilakukan penginisialisasian matriks xy dengan nilai awal **x** dan **y** yang sudah ditentukan sebelumnya.

```
Command Window
>> f = lagrange(xy)
f =

@(x) (x - 7.81) ./ (-2.56) .* (x - 9.2) ./ (-3.95) .* 6 + (x - 5.25) ./ (2.56) .
- 9.2) ./ (-1.39) .* 5 + (x - 5.25) ./ (3.95) .* (x - 7.81) ./ (1.39) .* 3.5

>> f(7.5)
ans = 5.2427
>> |
```

Pada *command window* di atas, terdapat matriks xy yang berisi nilai nilai dari x dan y yang berjumlah 3 buah titik. Kemudian, nilai dari xy tersebut dimasukkan ke dalam parameter *function* **lagrange** yang hasilnya dimasukkan ke dalam f dimana f akan menghasilkan fungsi *anonymous* akan digunakan dalam perhitungan dengan rumus sebagai berikut.

$$y(x) = \left(\frac{x - x_1}{x_0 - x_1}\right)\left(\frac{x - x_2}{x_0 - x_2}\right)y(x_0) + \left(\frac{x - x_0}{x_1 - x_0}\right)\left(\frac{x - x_2}{x_1 - x_2}\right)y(x_1) \\ + \left(\frac{x - x_0}{x_2 - x_0}\right)\left(\frac{x - x_1}{x_2 - x_1}\right)y(x_2)$$

Setelah itu, nilai $x = 7.5$ yang dicari dimasukkan ke dalam f(x) dan **didapatkan hasilnya sebesar 5.2427**

2. ANALISIS PRAKTIKUM NEWTON DIVIDED-DIFFERENCE

```
Command Window
>> xy = [5.25 6.0;7.81 5.0;9.2 3.5]
xy =

    5.2500    6.0000
    7.8100    5.0000
    9.2000    3.5000
```

Dilakukan penginisialisasian matriks xy dengan nilai awal **x** dan **y** yang sudah ditentukan sebelumnya.

```
Command Window
>> f = ndd(xy)
f =

@(x) 6 + -0.39063 .* (x - 5.25) + -0.17431 .* (x - 5.25) .* (x - 7.81)

>> f(7.5)
ans = 5.2427
>>
```

Pada *command window* di atas, terdapat matriks xy yang berisi nilai nilai dari x dan y yang berjumlah 3 buah titik. Kemudian, nilai dari xy tersebut dimasukkan ke dalam parameter *function* **ndd** yang hasilnya dimasukkan ke dalam f dimana f akan menghasilkan fungsi *anonymous* akan digunakan dalam perhitungan dengan rumus sebagai berikut.

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

Setelah itu, nilai $x = 7.5$ yang dicari dimasukkan ke dalam f(x) dan **didapatkan hasilnya sebesar 5.2427**