LABORATORIUM KOMPUTASIONAL
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS YARSI

METODE NUMERIK
Modul IV
"Interpolation"

a. **Estimasi waktu**: 100 menit

b. **Tujuan Istruksional Khusus**:
- Mahasiswa dapat menggunakan *Mathlab* dengan baik untuk memecahkan permasalahan numerik
- Mahasiswa dapat memahami pembuatan fungsi dalam Matlab
- Mahasiswa dapat mencari solusi Interpolasi dari beberapa titik koordinat.

c. **Landasan Teori**: -

Misalkan diberikan n + 1 titik data $(x_1, y_1), \ldots, (x_n, y_n)$, dengan $x_1, \ldots, x_n$ semuanya berbeda. Misalkan $p_n(x)$ adalah polinom berderajat ≤ n yang melewati semua titik tersebut. Polinom interpolasi Newton dari data tersebut dapat dituliskan dalam bentuk

$$p_n(x) = a_1 + a_2(x - x_1) + a_3(x - x_2)(x - x_1) + \ldots + a_n(x - x_1)\ldots(x - x_{n-1})$$

Untuk mencari konstanta a_1, subtitusikan nilai $p_n(x)$ untuk $x = x_1$.

$$p_n(x_1) = f(x_1) + a_1(x_2 - x_1) = f(x_1)$$

$$a_2 = \frac{f(x_2) - f(x_1)}{(x_2 - x_1)}$$

Sekarang kita definisikan hal-hal sebagai berikut :

$$f[x_k] = f(x_k)$$

$$f[x_k, x_{k+1}] = \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}$$

$$f[x_k, x_{k+1}, x_{k+2}] = \frac{f[x_{k+1}, x_{k+2}] - f(x_k, x_{k+1})}{x_{k+2} - x_k}$$

$$\vdots$$

$$f[x_k, x_{k+1}, \cdots, x_n] = \frac{f[x_{k+1}, \cdots, x_n] - f(x_k, x_{n-1})}{x_n - x_k}$$

Persamaan di atas disebut *persamaan beda terbagi*.

Dari notasi beda terbagi ini kita dapat menuliskan $a_1 = f[x_0, x_1]$. Konstanta yang lain dapat diturunkan dengan cara yang sama, tetapi karena terlalu panjang, tidak diberikan di sini. Selanjutnya dapat digunakan sutu kesimpulan bahwa

$a_k = f[x_0, x_1, \cdots, x_k]$, untuk k = 0, 1, . . . , n

Sehingga $p_n(x)$ dapat dituliskan sebagai

$$p_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \cdots, x_n](x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

Rumus ini sering disebut pula sebagai polinom interpolasi beda terbagi Newton. Sebagai catatan, untuk k ≥ 0 berlaku

$$p_{k+1}(x) = p_k[x] + (x - x_0)(x - x_1)\cdots(x - x_k) f[x_0, x_1, \cdots, x_k - x_{k+1}]$$

Hal ini merupakan salah satu kelebihan dari polinom interpolasi Newton dibandingkan polinom interpolasi lagrange. Dengan penambahan satu buah titik data, kita tinggal menghitung ruas terakhir saja. (*coba anda perhatikan kedua rumus tersebut* !).

d. **Langkah-langkah**:

1. Buka salah program matlab.
2. Pilih menu File → New → M.File
   Buat suatu program sederhana membuat fungsi polinomial Newton
   $P_{k-1}(z) = a_1 + a_2(z-x_1) + ... + a_k(z-x_1)(z-x_2)..(z-x_{k-1})$

   Ketiklah di dalam M. File editor

```
function P=nilaiPolaN(a,x,z)
%untuk menghitung nilai polinomial
%P{k-1}(z)=a1+a2(z-x1)+a3(z-x1)(z-x2)+ ... + ak(z-x1)(z-x2)...(z-xk-1)
k=length(a);
P=a(1);
for j=2:k,
   P=P+a(j)*prod(z-x(1:j-1));
end
```

   simpanlah file tersebut dengan nama file nilaiPolaN. Fungsi Polinomial Newton telah terbuat.
3. Selanjtunya buatlah fungsi interpolasi untuk mencari nilai a1, a2, ..., an. Ketiklah

```
function a=interpolaN(x,y)
% Untuk menghitung a1, a2, ...,an pada
%P{n-1)=a1 + a2(x-x1)+a3(x-x1)(x-x2)+ ...an(x-x1)(x-x2)(x-x{n-1)
```

```
%yang melalui titik (x1,y1),(x2,y2),(xn,yn)
n=length(x);
a(1)=y(1);
for k=2:n,
   P=nilaiPolaN(a(1:k-1),x(1:k-1),x(k));
   M=prod(x(k)-x(1:k-1));
   a(k)=(y(k)-P)/M;
end
```

simpanlah file tersebut dengan nama file interpolaN

4. Ujilah program tersebut, carilah nilai a1, a2, ... an jika diketahui koordinat titik :

x	0	1	-1	2	-2
f(x)	-5	-3	-15	39	-9

pada command windows ketiklah

```
>> x=[0 1 -1  2 -2]
>>y=[-5  -3  -15  -39  -9]
>>a=interpolaN(x,y)
>> ??????
```

5. Carilah untuk koordinat titik lain misal

x	1	2	5	7
f(x)	2	9	126	344

x	1	2	3	4	5	6
f(x)	-3	0	15	48	105	192

Selamat mencoba. !!!