

# STRUKTUR REKURSIF

Pertemuan

6

## Rekursif

adalah suatu proses yang bisa memanggil dirinya sendiri.



Daun Pakis



Pohon Cemara

Daun Pakis dibentuk oleh ranting-ranting daun yang mempunyai pola yang mirip dengan daun pakis itu sendiri. Setiap ranting daun disusun lagi oleh ranting daun dengan pola yang mirip. Demikian juga dengan pohon cemara. Objek yang mempunyai pola rekursif ini disebut fraktal.

## STRUKTUR REKURSIF

Contoh konsep penggunaan Rekursif

**Masalah :** Memotong Roti tawar tipis-tipis sampai habis

**Algoritma :**

1. Jika roti sudah habis atau potongannya sudah paling tipis maka pemotongan roti selesai.

2. Jika roti masih bisa dipotong, potong tipis dari tepi roti tersebut, lalu lakukan prosedur 1 dan 2 untuk sisa potongannya.

### Contoh Fungsi Rekursif

#### a. Fungsi pangkat

- Fungsi ini digunakan untuk menghitung nilai:  $X^n$  dengan  $n$  berupa bilangan bulat positif.  
Solusi dari persoalan ini:

- JIKA  $n = 1$  MAKA  $X^n = X$

- SELAIN ITU:  $X^n = X * X^{n-1}$

Algoritma Fungsi Pangkat:

Berikut adalah fungsi pangkat dengan menggunakan solusi di atas:

```
int pangkat(int a, int b){
    if (b==1){
        return a;
    }
    else {
        return a*pangkat(a,b-1);
    }
}
```

#### b. Faktorial

$$0! = 1$$

$$N! = N \times (N-1)! \text{ Untuk } N > 0$$

Scr notasi pemrograman dapat ditulis sebagai :

$$\text{FAKT}(0) = 1 \dots\dots\dots (1)$$

$$\text{FAKT}(N) = N * \text{FAKT}(N-1) \dots\dots\dots (2)$$

Contoh :

hitung 5!, dengan menggunakan secara rekursif

5! Adalah :

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

**Dengan menggunakan program**

```
#include <iostream.h>
#include <iomanip.h>
#include <conio.h>
unsigned long factorial (unsigned long number);
main()
{ for (int i=0; i<=10; i++)
cout << setw(2) << i << "!" << factorial(i) << endl;
//return 0;
getch();}
// recursive definition of function factorial unsigned long factorial (unsigned long
number)
{ if (number <=1) // base case return 1;
else
return number * factorial(number-1); }
```

### c. Fibonancy

Deret Fibonaccy adalah deret dimana nilai suku ke n merupakan jumlah nilai suku ke n-1 dan suku ke n-2.

Untuk suku  $n > 2$ ,

Suku pertama ( $n=1$ ) nilainya adalah 1, dan suku ke dua ( $n=2$ ) nilainya = 1.

Hasil deret fibonaccy adalah sebagai berikut: Deret Fibonaccy :

0,1,1,2,3,5,8,13,.....

Secara notasi pemrograman dapat ditulis sebagai :

$$\text{Fibo (1)} = 0 \ \& \ \text{Fibo (2)} = 1 \ \dots\dots\dots (1)$$

$$\text{Fibo (N)} = \text{Fibo (N-1)} + \text{Fibo (N-2)} \ \dots\dots\dots (2)$$

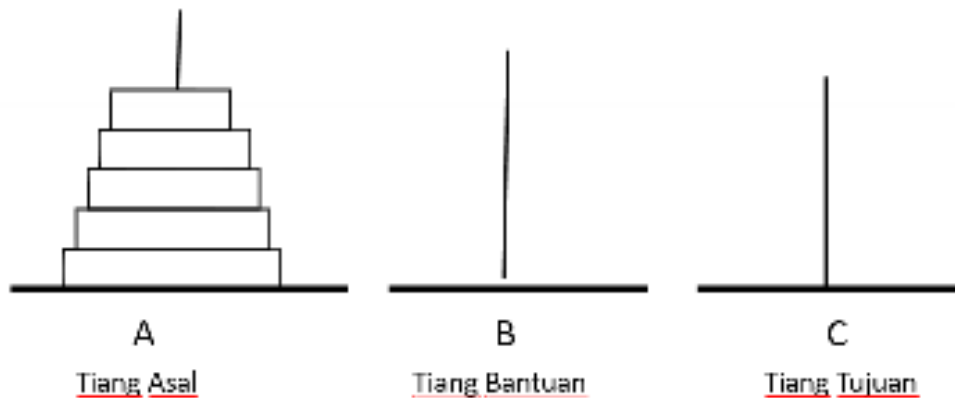
Dengan menggunakan program deret fibonancy dapat di tuliskan sebagai berikut :

```
A[1] = 1; A[2] = 2;
For (i=3; i<=10; i++)
{
    A[i] = A[i-1] + A[i-2];
}
```

**d. Menara Hanoi**

- Legenda klasik pendeta Budha di kota Hanoi (Vietnam)
- Terdapat tiga buah tiang tegak setinggi 5 meter dengan 64 buah piringan (disk).
- Setiap piringan mempunyai lubang ditengahnya agar dapat dimasukkan kedalam tiang.
- Bagaimana cara memindahkan seluruh piringan tersebut ke sebuah tiang yang lain, setiap kali hanya satu piringan yang boleh dipindahkan.
- Syarat: tidak boleh ada piringan besar diatas piringan yang lebih kecil.

Konsep menara Hanoi



Rumus Langkah Pemindahan :

$$2^N - 1$$

N = Jumlah Piringan