

Struktur Data dan Algoritma “Rekursif”



Oleh:

Indra Hermawan

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

Rekursif

- Proses yang memanggil dirinya sendiri.
- Merupakan suatu fungsi atau prosedur
- Terdapat suatu kondisi untuk berhenti.

Faktorial

Konsep Faktorial

$$n! = n(n-1)(n-2)\dots 1$$

Dapat diselesaikan dengan

Cara Biasa —

Rekursif

$$F(n) = \begin{cases} 1 & \text{jika } n=0, n=1 \\ n * F(n) & \text{jika } n > 1 \end{cases}$$

Faktorial : Cara Biasa

```
Int Faktorial(int n)
{
    if (n<0) return -1 ;
    else if (n>1)
    {
        S = 1 ;
        for(i=2 ;i<=n;i++) S = S * n ;
        return S ;
    }
    else return 1 ;
}
```

Int Faktorial(int n)

```
{  
    if (n<0) return -1  
    else if (n>1) Return (n*Faktorial(n-1))  
    Else Return 1 ;  
}
```

Deret Fibonacci

- ❑ Leonardo Fibonacci berasal dari Italia 1170-1250
- ❑ Deret Fibonacci f_1, f_2, \dots didefinisikan secara rekursif sebagai berikut :
 - $f_1 = 1$
 - $f_2 = 2$
 - $f_n = f_{n-1} + f_{n-2}$ for $n \geq 3$
- ❑ Deret: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, ...

Deret Fibonacci

```
procedure fab(n)
```

```
    if n=1 then  
        return 1
```

```
    if n=2 then  
        return 2
```

```
    return (fab(n-1) + fab(n-2))
```

```
end
```

Rekursif Tail

- Jika pernyataan terakhir yang akan dieksekusi berada dalam tubuh fungsi
- Hasil yang kembali pada fungsi tsb bukanlah bagian dari fungsi tersebut.
- Tidak memiliki aktivitas selama fase balik.

Rekursif Tail : Faktorial()

$$F(n,a) = \begin{cases} a & \text{jika } n=0, n=1 \\ F(n-1,na) & \text{jika } n>1 \end{cases}$$

$$F(4,1) = F(3,4)$$

Fase awal

$$F(3,4) = F(2,12)$$

$$F(2,12) = F(1,24)$$

$$F(1,24) = 24$$

Kondisi Terminal

24

Fase Balik
Rekursif Lengkap

Algoritma BinRec(n)

//input : Bilangan desimal integer positif n

//output : Jumlah digit biner yang dinyatakan dengan n

If (n=1) return 1

Else return BinRec($\lfloor n/2 \rfloor$) + 1