

CCH1A4 / Dasar Algoritma & Pemrograman

Yuliant Sibaroni M.T, Abdurahman Baizal M.Kom

KK Modeling and Computational Experiment



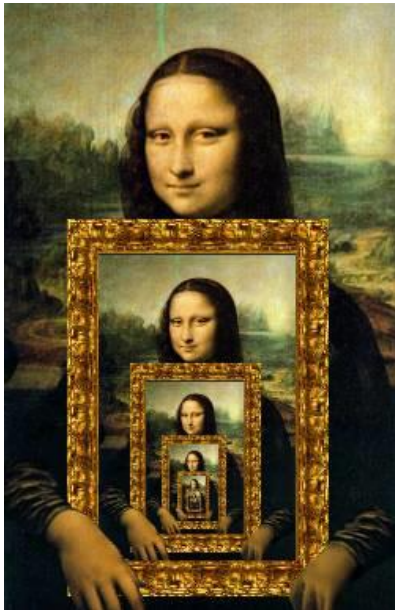
Rekursif

- ▶ Definisi
- ▶ Fungsi iterative dan rekursif
- ▶ Prosedur iterative dan rekursif

Definisi

Berfikir Rekursif

Memecahkan masalah besar dengan menjadikannya submasalah yang lebih kecil. **Rekursif** >< **Iterative**



Definisi

Konsep Rekursif

Konsep rekursif dapat diterapkan dalam sebuah fungsi atau prosedur. Dengan rekursif, memungkinkan bagi sebuah fungsi/prosedur untuk **memanggil dirinya** sendiri dalam bagian algoritmanya.

Struktur

Ada 2 bagian utama pada fungsi/prosedur yang menggunakan konsep rekursif didalamnya, yaitu :

- ▶ **Bagian Basis**

Berisi **kondisi berhenti** dari proses rekursif, dan melakukan aksi yang sesuai untuk kondisi tersebut

- ▶ **Bagian Rekursion/Rekuren**

Berisi perintah untuk pemanggilan dirinya sendiri

Fungsi

Contoh 13.1 Fungsi Iterative

Pembuatan fungsi untuk menghitung jumlah bilangan $1+2+\dots+n$, dengan input bilangan = n

```
function Jumlah(n: integer) → integer
```

Kamus lokal

```
  i, hasil : integer
```

Algoritma

```
  hasil ← 0
```

```
  for i ← 1 to n do
```

```
    hasil ← hasil+i
```

```
  → hasil
```

Fungsi

Contoh 13.2 Fungsi Rekursif

Pembuatan fungsi untuk menghitung jumlah bilangan $1+2+\dots+n$, dengan input bilangan = n

Tahapan Berfikir

- ▶ Cari rumus untuk Jumlah(n)
 - Jumlah(1) = 1
 - Jumlah(2) = 1+2
 - Jumlah(3) = 1+2+3
 - ...
 - Jumlah($n-1$) = 1+2+.....+($n-1$)
 - Jumlah(n) = 1+2+.....+($n-1$) + n
 - = jumlah($n-1$) + n

$$Jumlah(n) = \begin{cases} 1 & n = 1 \\ n + Jumlah(n-1) & n > 1 \end{cases}$$



Basis



Rekuren

Fungsi

Contoh 13.2 Fungsi Jumlah Rekursif

Fungsi penentuan jumlah $(1+2+\dots+n)$

```
Function Jumlah(n)  $\rightarrow$  integer
```

```
{asumsi : n  $\geq$  1}
```

```
Algoritma
```

```
  if (n=1) then
```

```
     $\rightarrow$  1
```

```
  else
```

```
     $\rightarrow$  n + Jumlah(n-1)
```

Fungsi

Contoh 13.3 Fungsi Fibonacci Rekursif

Fungsi penentuan suku ke- n Barisan Fibonacci

Barisan Fibonacci : 1,1,2,3,5,8,13,...

Rumus

- ▶ $Fibbo(1) = 1$
- ▶ $Fibbo(2) = 1$
- ▶ $Fibbo(3) = 1 + 1 = 2 = Fibbo(1) + Fibbo(2)$
- ▶ $Fibbo(4) = 1 + 2 = 3 = Fibbo(2) + Fibbo(3)$
- ▶ $Fibbo(5) = 2 + 3 = 5 = Fibbo(3) + Fibbo(4)$
- ▶ ...
- ▶ $Fibbo(n) = Fibbo(n-2) + Fibbo(n-1)$

$$Fibbo(n) = \begin{cases} 1 & n = 1 \text{ or } n = 2 \\ Fibbo(n-2) + Fibbo(n-1) & n > 2 \end{cases}$$

Fungsi

Contoh 13.3 Fungsi Fibonacci Rekursif

Fungsi suku deret Fibonacci secara Rekursif

```
Function Fibbo(n)  $\rightarrow$  integer  
{asumsi : n >= 1}  
Algoritma  
  if (n=1) or (n=2) then  
     $\rightarrow$  1  
  else  
     $\rightarrow$  Fibbo(n-2) + Fibbo(n-1)
```

Fungsi

Contoh 13.4 Fungsi Faktorial Rekursif

Fungsi untuk menghitung nilai $n!$ secara rekursif

Rumus

- ▶ Faktorial(0) = 1
- ▶ Faktorial(1) = 1
- ▶ Faktorial(2) = 2×1
- ▶ Faktorial(3) = $3 \times 2 \times 1 = 3 \times \text{Faktorial}(2)$
- ▶ Faktorial(4) = $4 \times 3 \times 2 \times 1 = 4 \times \text{Faktorial}(3)$
- ▶ ...
- ▶ Faktorial(n) = $n \times (n-1) \times \dots \times 3 \times 2 \times 1 = n \times \text{Faktorial}(n-1)$

$$Faktorial(n) = \begin{cases} 1 & n = 0 \text{ or } n = 1 \\ n \times Faktorial(n-1) & n > 1 \end{cases}$$

Fungsi

Contoh 13.4 Fungsi Faktorial Rekursif

Fungsi perhitungan $n!$

```
Function Faktorial(n)  $\rightarrow$  integer
```

```
{asumsi :  $n \geq 0$ }
```

Kamus Lokal

Algoritma

```
if (n=0) or (n=1) then
```

```
     $\rightarrow$  1
```

```
else
```

```
     $\rightarrow$  n * Faktorial(n-1)
```

Prosedur

Contoh 13.5

Prosedur Cetak(i), untuk menampilkan data $X[1]$ sampai dengan $X[i]$
(X :array[1..100] of integer)

Basis

Output($X[i]$) untuk $i=1$

Rekursion

Output($X[i]$); Cetak($i-1$), untuk $i>1$

Prosedur

Contoh 13.5 Prosedur Cetak1(i)

Menampilkan nilai X[i] sampai dengan X[1]

```
procedure Cetak(input i:integer)  
{asumsi : i >= 1}  
Kamus Lokal  
Algoritma  
  if (i=1) then  
    output(X[i])  
  else  
    output(X[i])  
    Cetak(X[i-1])
```

Prosedur

Contoh 13.6 Prosedur Cetak2(i)

Menampilkan nilai X[i] sampai dengan X[100]

```
procedure Cetak(input i:integer)  
{asumsi : i >= 1}  
Kamus Lokal  
Algoritma  
  if (i=100) then  
    output(X[i])  
  else  
    output(X[i])  
    Cetak(X[i+1])
```

Latihan

Soal

1. Buat fungsi hitung1(n) secara rekursif, bila diketahui:

$$\text{Hitung1}(1) = 2$$

$$\text{Hitung1}(2) = 5$$

$$\text{Hitung1}(3) = 3 \times (2+5) = 21$$

$$\text{Hitung1}(4) = 4 \times (5+21) = 109$$

dst

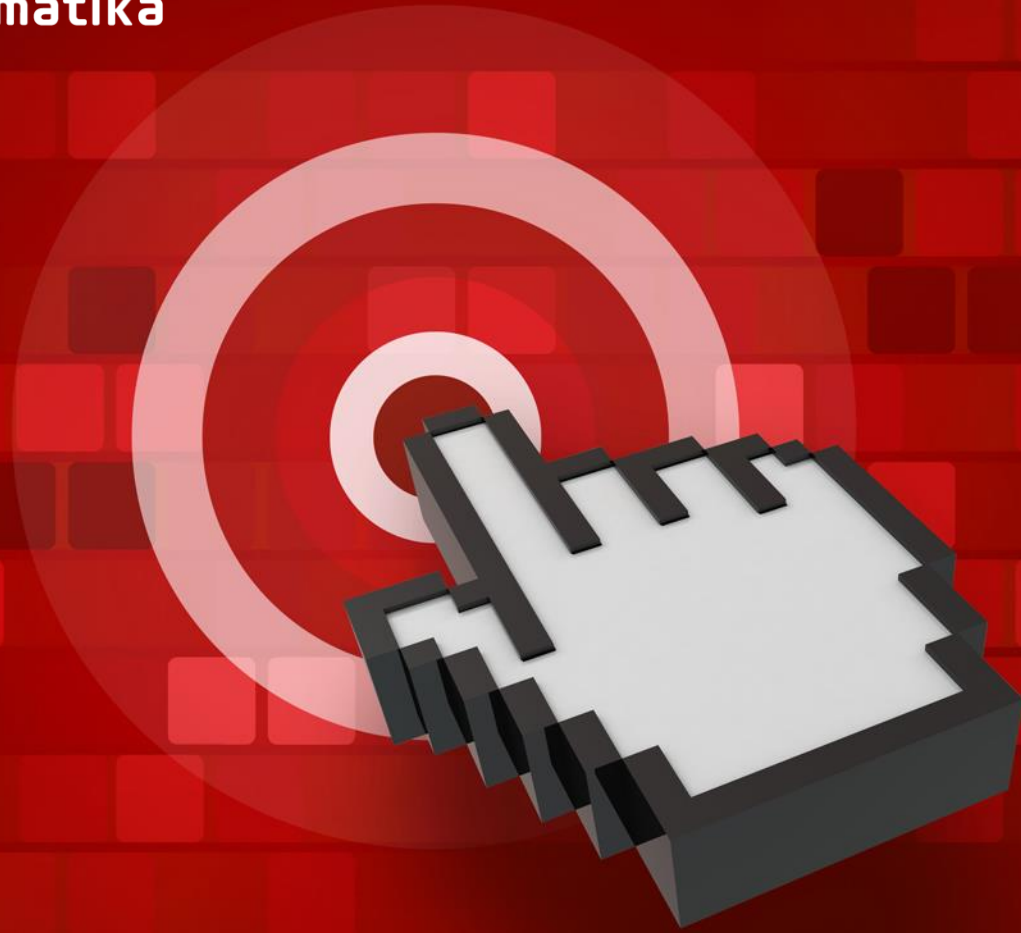
2. Buat fungsi hitung2(n) secara rekursif yang akan menghitung jumlah bilangan ganjil sampai dengan n (n bisa genap / ganjil)
3. Buat fungsi hitung3(a,b) secara rekursif yang akan menghitung jumlah kuadrat dari a sampai dengan b. Misal a=2, b=5 maka $\text{hitung3}(2,5) = 2^2 + 3^2 + 4^2 + 5^2$

Pustaka

- ▶ Inggriani Liem, Diklat Kuliah IF223 Algoritma Dan Pemrograman, Jurusan Teknik Informatika Bandung, 1999
- ▶ Rinaldi Munir, Algoritma dan Pemrograman dalam bahasa Pascal dan C, edisi ke-3, penerbit Informatika 2004



Fakultas Informatika
School of Computing
Telkom University



THANK YOU