

Pertemuan X & XI

- KONSEP MODULAR

- Pengertian
 - ✓ Konsep Modular memecah program ke dalam modul-modul.
Contoh modul yang sering digunakan adalah : **Prosedur & Fungsi**
- Variabel

Ada dua jenis variabel yang dikenal dalam konsep modular, yaitu :

 - ✓ *Variabel Lokal*
Variabel yang dideklarasikan di dalam modul. Hanya dapat digunakan di dalam modul tersebut.
 - ✓ *Variabel Global*
Variabel yang dideklarasikan di dalam program utama. Dapat digunakan di seluruh badan program, termasuk di dalam modul.
- Parameter
 - ✓ Pengertian
Parameter adalah struktur yang digunakan untuk menunjukkan proses pertukaran variabel dari dan ke dalam modul.
 - ✓ Jenis – Jenis Parameter
 - ❖ **Input** (*by value*)
Parameter yang menunjukkan variabel yang masuk ke modul. Variabel yang dideklarasikan di luar modul (program utama), dan dipakai di dalam modul.
 - ❖ **Output** (*by reference*)
Parameter yang menunjukkan variabel yang keluar dari modul. Variabel yang dideklarasikan di dalam modul, dan dipakai di luar modul (program utama).
 - ❖ **Input/Output** (*by reference*)
Parameter yang menunjukkan variabel yang masuk ke dan keluar dari modul. Variabel yang dideklarasikan di luar modul (program utama), kemudian dipakai di dalam dan di luar modul.

Modul	PROSEDUR	FUNGSI
Nilai Kembali (return)	Tidak Ada	Ada
Parameter yg digunakan	Input, Output, I/O	Input

- PROSEDUR

- Format :

Procedure namaprosedur (parameter, jika ada)
 {Spesifikasi Prosedur}
 {I.S : Keadaan Awal Prosedur}
 {F.S : Keadaan Akhir Prosedur}

DEKLARASI
 { variabel lokal }

ALGORITMA
 { urutan instruksi dalam prosedur }

- Contoh Kasus :

✓ Menghitung Rata-rata

Pseudocode

PROGRAM prosedur1
 {I.S : Banyak bilangan diinputkan}
 {F.S : Rata-rata bilangan ditampilkan}
DEKLARASI
 N : integer
 ratarata : real
procedure hitung_rata2 (input N : integer, output rata2 : real)

ALGORITMA
 write('Berapa banyak bilangan? ')
 read(N)
 hitung_rata2(N, rata2)
 write('Rata-rata = ', rata2)

Procedure hitung_rata2 (input N : integer, output rata2 : real)
 {Menghitung rata2}
 {I.S : Banyak bilangan terdefinisi}
 {F.S : Hasil perhitungan rata-rata bilangan diperoleh}

DEKLARASI
 { variabel lokal }
 bilangan, i, jumlah : integer

ALGORITMA

```
{ urutan instruksi dalam prosedur }  
jumlah  $\leftarrow$  0  
for i  $\leftarrow$  1 to N do  
    read(bilangan)  
    jumlah  $\leftarrow$  jumlah + bilangan  
endfor  
rata2  $\leftarrow$  bilangan/N
```

PASCAL

PROGRAM prosedur1;

Var

 N : integer;

 rata2 : real;

procedure hitung_rata2 (N : integer; **var** rata2 : real);

 {Menghitung rata2}

 {I.S : Banyak bilangan terdefinisi}

 {F.S : Hasil perhitungan rata-rata bilangan diperoleh}

 {DEKLARASI}

 {variabel lokal}

Var

 bilangan, i, jumlah : integer;

 {ALGORITMA}

 {urutan instruksi dalam prosedur}

begin

 jumlah := 0;

for i := 1 **to** N **do**

begin

 write('Bilangan ',i,' '); readln(bilangan);

 jumlah := jumlah + bilangan;

end;

 rata2 := jumlah / N;

end;

begin

 write('Berapa banyak bilangan? '); readln(N);

 hitung_rata2(N, rata2);

 write('Rata-rata = ', rata2:0:2);

 readln();

end.

- **FUNGSI**

- Format :

Function namafungsi (parameter, jika ada) → tipe_kembalian
{Spesifikasi Fungsi}
{I.S : Keadaan Awal Fungsi}
{F.S : Keadaan Akhir Fungsi}

DEKLARASI

{ variabel lokal }

ALGORITMA

{ urutan instruksi dalam fungsi }

- Contoh Kasus :

- ✓ **Menghitung Rata-rata**

Pseudocode

PROGRAM fungsi1

{I.S : Banyak bilangan diinputkan}
{F.S : Rata-rata bilangan ditampilkan}

DEKLARASI

N : integer

ratarata : real

fungsi hitung_rata2 (input N : integer)

ALGORITMA

write('Berapa banyak bilangan? ')

read(N)

write(hitung_rata2(N))

Function hitung_rata2 (input N : integer)

{Menghitung rata2}

{I.S : Banyak bilangan terdefinisi}

{F.S : Hasil perhitungan rata-rata bilangan diperoleh}

DEKLARASI

{ variabel lokal }

bilangan, i, jumlah : integer

ALGORITMA

{ urutan instruksi dalam fungsi }

jumlah ← 0

for i ← 1 to N do

 read(bilangan)

 jumlah ← jumlah + bilangan

endfor

```
rata2 ← bilangan/N  
return rata2
```

PASCAL

PROGRAM fungsi1;

Var

N : integer;

rata2 : real;

function hitung_rata2 (N : integer);

{Menghitung rata2}

{I.S : Banyak bilangan terdefinisi}

{F.S : Hasil perhitungan rata-rata bilangan diperoleh}

{DEKLARASI}

{variabel lokal}

Var

bilangan, i, jumlah : integer;

{ALGORITMA}

{urutan instruksi dalam fungsi}

begin

jumlah := 0;

for i := 1 **to** N **do**

begin

write('Bilangan ',i,' = '); readln(bilangan);

jumlah := jumlah + bilangan;

end;

hitung_rata2 := jumlah / N; {return}

end;

begin

write('Berapa banyak bilangan? '); readln(N);

write(hitung_rata2(N):0:2);

readln();

end.