

FUNGSI (*FUNCTION*)

Definisi Fungsi

- Berbeda dengan prosedur yang didefinisikan dengan kata kunci **function**. Berikut ini bentuk umum dari pendefinisian sebuah fungsi.

```
Function NamaFungsi (parameter1 : tipe_data, parameter2: tipe_data, . . .) : tipe_data;  
const  
    {daftar konstanta lokal}  
Var  
    {daftar pendeklarasian variabel lokal}  
Begin  
    {kode program yang akan ditulis}  
. . .  
NamaFungsi := nilai_kembalian;    {ingat baris ini}  
End;
```

Definisi Fungsi

- Berikut ini contoh pendefinisian fungsi sederhana yang akan mengalikan dua buah bilangan bulat.

```
Function Kali (x, y : integer) : longint;  
Begin  
    Kali := x * y;  
End;
```

- Kita juga dapat mendeklarasikan variabel lokal untuk menyimpan nilai sebelum akhirnya dikembalikan lagi ke fungsi, misalnya seperti kode di bawah ini.

```
Function Kali (x, y : integer) : longint;  
Var  
    Hasil : longint;  
Begin  
    Hasil := x * y;  
    Kali := Hasil;  
End;
```

Pemanggilan Fungsi

- Fungsi diakses dengan cara memanggil namanya dari program pemanggil, diikuti dengan daftar parameter aktual (bila ada). Oleh karena fungsi menghasilkan sebuah nilai maka pada saat pemanggilannya juga dapat ditampung ke dalam suatu variabel.
- Berikut ini contoh pemanggilan fungsi di atas.

Var

A : longint;

Begin

{memanggil fungsi Kali dan menyimpan nilainya ke dalam variabel A}

A := Kali (10, 5);

...

End.

Contoh 1 :

Program Maksimum;

Deklarasi

a,b : integer;

Function Maks(x,y : integer):integer;

Deklarasi

{Tidak ada}

Algoritma

if X < Y then maks = y else maks = x

endif

Algoritma

Read(A,B)

Write(Maks(a,b))

Program GenapGanjil;

Var

X : integer;

Function Genap(n : integer) : boolean;

Begin

Genap = (n mod 2 = 0)

end

Begin

Read(x)

if Genap(x) then write('Genap') else
write('Ganjil')

endif

Contoh 2

- Buatlah fungsi untuk menentukan apakah sebuah tahun merupakan tahun kabisat atau bukan. Suatu tahun disebut tahun kabisat jika tahun tersebut habis dibagi 4 atau habis dibagi 400 jika tahun tersebut kalipatan 100!

```
Function Kabisat(input th : integer) boolean
```

```
DEKLARASI
```

```
{tidak ada}
```

```
ALGORITMA
```

```
if (th mod 4 = 0 and th mod 100  $\neq$  0) or (th mod 400 = 0) then  
    return true {tahun kabisat}
```

```
Else
```

```
    return false [bukan tahun kabisat]
```

```
endif
```

Contoh 3 :

- Tulislah fungsi untuk menentukan nama-nama bulan berdasarkan nomor bulannya (1 sampai 12)

```
Function NamaBulan(input bln : integer)    string
{mengembalikan nama bulan berdasarkan nomor bln}
```

```
DEKLARASI
```

```
nama : string
```

```
ALGORITMA
```

```
case bln
```

```
1 : nama      `Januari`
```

```
2 : nama      `Februari`
```

```
3 : nama      `Maret`
```

```
4 : nama      `April`
```

```
end case
```

```
return nama
```

Ket : Fungsi nama bulan (versi 1 : menggunakan peubah)

```
Function NamaBulan(input bln : integer)    string  
{mengembalikan nama bulan berdasarkan nomor bln}
```

```
DEKLARASI
```

```
    {tidak ada}
```

```
ALGORITMA
```

```
case bln
```

```
    1 : return    'Januari'
```

```
    2 : return    'Februari'
```

```
    3 : return    'Maret'
```

```
    4 : return    'April'
```

```
end case
```

Ket : Fungsi nama bulan (versi 2 : tanpa menggunakan peubah)

Contoh 4 :

```
Function Hitung(Var A,B : integer): integer;
```

```
Begin
```

```
    Hitung := A + B;
```

```
End;
```

```
Var X,Y : integer;
```

```
Begin
```

```
    Write('Nilai X ? ');
```

```
    Readln(X);
```

```
    Write('Nilai Y ? ');
```

```
    Readln(Y);
```

```
    Writeln;
```

```
    Writeln(X, ' + ', Y, ' = ', Hitung(X,Y));
```

```
End.
```

Hasilnya :

Nilai X ? 2

Nilai Y ? 3

2 + 3 = 5

Latihan :

- Buat program perhitungan nilai faktorial menggunakan fungsi.

```
Program faktorial;
Var
n : integer;
function f(n:integer);integer;
var
i, f : integer;
begin
f:=1;
for i:=1 to n do
    f:=f +1;
end;
begin
write ('jumlah faktorial :');readln(n);
writeln(f);
end.
```