



PROSEDUR DAN FUNCTION

PROSEDUR DAN FUNCTION

PROSEDUR

Prosedur adalah suatu program yang terpisah dalam blok sendiri yang berfungsi sebagai subprogram (program bagian). Prosedur diawali dengan kata cadangan `PROCEDURE` di dalam bagian deklarasi prosedur. Prosedur dipanggil dan digunakan di dalam blok program lainnya dengan menyebutkan judul prosedurnya.

Bentuk penulisan :

```
PROGRAM judul-program;  
  PROCEDURE judul-prosedur;  
  BEGIN  
    . . .  
  END;
```

```
BEGIN  
  . . . . .  
END.
```



Contoh :

```
PROGRAM prosedur;  
USES CRT;  
PROCEDURE garis;  
BEGIN  
  WRITELN('-----');  
END;
```

```
BEGIN  
  CLRSCR;  
  garis;  
  WRITELN('BELAJAR PROSEDUR');  
  garis;  
  READLN;  
END.
```

Hasil :

```
-----  
BELAJAR PROSEDUR  
-----
```



PARAMETER DALAM PROSEDUR

Nilai di dalam suatu modul program Pascal sifatnya adalah lokal, artinya hanya dapat digunakan pada modul atau unit program yang bersangkutan saja, tidak dapat digunakan pada modul atau unit program yang lain.

PENGIRIMAN PARAMETER SECARA NILAI

Bila parameter dikirim secara nilai (by value), parameter formal di prosedur akan berisi nilai yang dikirimkan yang kemudian bersifat lokal di prosedur. Bila nilai parameter formal di dalam prosedur tersebut berubah, tidak akan mempengaruhi nilai parameter nyata.

Pengiriman nilai ini merupakan pengiriman searah yang tidak dikirimkan balik dari parameter formal ke parameter nyata. Parameter yang digunakan dengan pengiriman secara nilai ini disebut dengan parameter nilai (value parameter)



Contoh :

```
PROGRAM parameter_value;
USES CRT;
PROCEDURE hitung(A,B,C : INTEGER);
BEGIN
    C:=A+B;
    WRITELN('Nilai di Prosedur');
    WRITELN('NILAI A= ',A,' NILAI B= ',B,' NILAI C= ',C);
END;
VAR
    X,Y,Z : INTEGER;
BEGIN
    CLRSCR;
    WRITE('NILAI X= '); READLN(X);
    WRITE('NILAI Y= '); READLN(Y);
    WRITE('NILAI Z= '); READLN(Z);
    hitung(X,Y,Z);
    WRITELN('Nilai setelah Prosedur');
```



```
WRITELN('NILAI X= ',X,' NILAI Y= ',Y,' NILAI Z= ',Z);  
  READLN;  
END.
```

Hasil :

NILAI X= 3

NILAI Y= 4

Nilai parameter Z sebelum dan sesudah prosedur sama

NILAI Z= 5 Nilai di Prosedur

NILAI A= 3 NILAI B= 4 NILAI C= 7

Nilai setelah Prosedur

NILAI X= 3 NILAI Y= 4 NILAI Z= 5



PENGIRIMAN PARAMETER SECARA ACUAN

Bila pengiriman parameter secara acuan (by reference), maka perubahan-perubahan yang terjadi pada nilai parameter formal di prosedur akan mempengaruhi nilai parameter nyata. Parameter-parameter ini disebut dengan *variabel parameter* serta dideklarasikan di deklarasi prosedur dengan menggunakan kata cadangan VAR.

Contoh :

```
PROGRAM parameter_reference;
USES CRT;
PROCEDURE hitung(VAR A,B,C : INTEGER);
BEGIN
    C:=A+B;
    WRITELN('Nilai di Prosedur');
    WRITELN('NILAI A= ',A,' NILAI B= ',B,' NILAI C= ',C);
END;
```



```
VAR
  X,Y,Z : INTEGER;
BEGIN
  CLRSCR;
  WRITE('NILAI X= '); READLN(X);
  WRITE('NILAI Y= '); READLN(Y);
  WRITE('NILAI Z= '); READLN(Z);
  hitung(X,Y,Z);
  WRITELN('Nilai setelah Prosedur');
  WRITELN('NILAI X= ',X,' NILAI Y= ',Y,' NILAI Z= ',Z);
  READLN;
END.
```

Hasil :

NILAI X= 3

NILAI Y= 4

NILAI Z= 0

Nilai parameter Z sebelum dan sesudah prosedur berbeda

Nilai di Prosedur NILAI A= 3 NILAI B= 4 NILAI C= 7

Nilai setelah Prosedur

NILAI X= 3 NILAI Y= 4 NILAI Z= 7



PROSEDUR MEMANGGIL PROSEDUR YANG LAIN

Di dalam prosedur dapat memanggil prosedur yang lainnya.

Contoh :

```
PROGRAM panggil_prosedur;  
USES CRT;  
PROCEDURE pro1(x1 : INTEGER);  
BEGIN  
    x1:=x1*x1;  
    Writeln('Nilai X di Prosedur1 = ',x1);  
END;  
  
PROCEDURE pro2(x2 : INTEGER);  
BEGIN  
    Writeln('Nilai X di Prosedur2 = ',x2);  
    pro1(x2);  
END;
```



```
VAR
  X : INTEGER;
BEGIN
  CLRSCR;
  WRITE('Masukkan nilai X= '); READLN(x);
  pro2(x);
  READLN;
END.
```

Hasil :

```
Masukkan nilai X= 12
Nilai X di Prosedur2 = 12
Nilai X di Prosedur1 = 144
```



PROSEDUR TERSARANG

Prosedur tersarang (nested procedure) adalah prosedur yang berada di dalam prosedur yang lainnya.

Contoh :

```
PROGRAM nested_prosedur;  
USES CRT;
```

```
PROCEDURE kesatu;  
  PROCEDURE kedua;  
  BEGIN  
    WRITELN('Prosedur KEDUA ini ada di dalam prosedur KESATU');  
  END;
```

```
PROCEDURE ketiga;  
  BEGIN  
    WRITELN('Prosedur KETIGA ini juga ada di dalam prosedur  
    KESATU');  
  END;
```



```
BEGIN
  WRITELN('Ini Prosedur KESATU akan memanggil Prosedur
  KEDUA & KETIGA');
  kedua;
  ketiga;
END;
```

```
BEGIN
  CLRSCR;
  WRITELN('Ini program di modul utama akan memanggil
  Prosedur KESATU');
  kesatu;
  READLN;
END.
```

Hasil :

Ini program di modul utama akan memanggil Prosedur KESATU
Ini Prosedur KESATU akan memanggil Prosedur KEDUA & KETIGA
Prosedur KEDUA ini ada di dalam prosedur KESATU
Prosedur KETIGA ini juga ada di dalam prosedur KESATU



PROSEDUR MEMANGGIL DIRINYA SENDIRI

Prosedur memanggil dirinya sendiri merupakan suatu prosedur yang memanggil atau menggunakan prosedur itu juga. Proses dari suatu program yang memanggil dirinya sendiri dikenal dengan nama *recursion*. Proses ini membutuhkan lebih banyak memori dalam komputer.

Contoh :

```
PROGRAM rekursi_prosedur;
USES CRT;
VAR
  I : INTEGER;

PROCEDURE rekursi;
BEGIN
  I:=I+1;
  WRITELN('PASCAL');
  IF I < 10 THEN rekursi;
END;
```



```
BEGIN
  CLRSCR;
  I:=0; rekursi;
  READLN;
END.
```

Hasil :

```
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
PASCAL
```



LATIHAN

Buat program untuk menghitung gaji harian PT. XYZ dengan ketentuan:

Gaji pokok karyawan Rp. 20000/jam

Bila karyawan bekerja lebih dari 7 jam/hari maka kelebihannya dihitung lembur yang besarnya 1.5 dari gaji pokok

Untuk karyawan yang bekerja 8 jam/hari atau lebih akan mendapat tambahan uang makan sebesar Rp. 5000

Karyawan yang bekerja 9 jam/hari atau lebih akan mendapat uang transport lembur sebesar Rp. 4000

Program ini akan terdiri dari 4 buah prosedur : pokok, lembur, makan, jasa

Input : NIP, Nama, Jumlah jam kerja



Output :

LAPORAN GAJI HARIAN KARYAWAN
PT. XYZ

Bulan : April 2011

NIP Nama Gaji Pokok Lembur Uang makan Transport Lembur

