

PROCEDURE

Prosedur adalah suatu program terpisah dalam blok sendiri yang berfungsi sebagai subprogram (program bagian). Diawali dengan kata cadangan "Procedure" didalam bagian deklarasi prosedur. Prosedur banyak digunakan pada program yang terstruktur karena :

- merupakan penerapan konsep program modular, yaitu memecah-mecah program yang rumit menjadi program-program bagian yang lebih sederhana dalam bentuk prosedur-prosedur.
- Untuk hal-hal yang sering dilakukan berulang-ulang, cukup dituliskan sekali saja dalam prosedur dan dapat dipanggil atau dipergunakan sewaktu-waktu bila diperlukan.

Bentuk Umum :

```
PROGRAM judul_program ;  
  
PROCEDURE judul_prosedur ;  
    Begin  
        Statement prosedur ;  
        ...  
        ...  
        ...  
    End ;  
Begin  
    Statement program utama ;  
    ...  
    ...  
    ...  
end.
```

1.1 Parameter Dalam Prosedur

Nilai didalam modul program Pascal sifatnya adalah local, artinya hanya dapat digunakan pada modul atau unit program yang bersangkutan saja, tidak dapat digunakan pada modul atau unit program yang lainnya.

Contoh program :

```
Procedure Tanya_hitung ;
Var
    X, Y : real ;
Begin
    Write ('Nilai X =') ;
    Readln (X) ;
    Y := X * X ;
End ;
Begin
    Tanya_hitung ;
    Writeln ('Nilai Y =', Y : 6 : 2 ) ;
End.
```

Keterangan contoh program :

Bila program dikompilasi, akan di deteksi kesalahan oleh compiler, karena variable Y yang sifatnya local di procedure Tanya_hitung digunakan pada program utama.

Agar nilai-nilai variable dapat digunakan di modul lainnya yang membutuhkan, maka dapat dilakukan dengan beberapa cara, yaitu :

- Dibuat bersifat global.
Harus di deklarasikan diatas modul yang akan menggunakannya.

Contoh :

```
Procedure kesatu ;
Begin
    ...
End ;
Var
    A, B : word ;
Procedure kedua ;
Begin
    ...
End ;
Procedure ketiga ;
Begin
    ...
End ;
Begin { program utama }
    ...
End.
```

Keterangan contoh :

Variabel A dan B dapat digunakan untuk modul Procedure kedua dan ketiga serta modul/program utama, tetapi tidak bersifat global untuk procedure kesatu.

- Dikirimkan sebagai parameter ke modul yang membutuhkannya. Parameter yang dikirimkan dari modul utama ke modul prosedur disebut dengan parameter nyata (actual parameter) dan parameter yang ada dan dituliskan pada judul prosedur disebut dengan parameter formal (formal parameter). Proses pengiriman data lewat parameter nyata ke parameter formal disebut dengan parameter passing. Parameter nyata dan parameter formal harus dengan tipe yang sama. Didalam pascal, parameter dapat dikirimkan secara nilai (by value) atau secara acuan (by reference).

1.1.1 Pengiriman Parameter Secara Nilai

Bila parameter dikirimkan secara nilai nilai, parameter formal di prosedur akan berisi nilai yang dikirimkan yang kemudian bersifat local di prosedur. Bila nilai parameter formal di prosedur berubah, tidak akan mempengaruhi nilai parameter nyata (nilai parameter nyata tetap, tidak berubah). Pengiriman secara nilai ini merupakan pengiriman searah, yaitu dari parameter nyata ke parameter formal, yang tidak dikirimkan balik dari parameter formal ke parameter nyata.

Parameter – parameter yang digunakan dengan pengiriman secara nilai ini disebut dengan parameter nilai (value parameter).

Contoh program :

```
Procedure Hitung(A, B : integer ) ;
Var
  C : integer ;
Begin
  C := A + B ;
  Writeln ( 'Nilai C =', C ) ;
End ;
Var
  X, Y : integer ;
Begin
  Write ( 'Nilai X =' ) ; readln ( X ) ;
  Write ( 'Nilai Y =' ) ; readln ( Y ) ;
  Hitung ( X, Y ) ;
End.
```

Output program :

Nilai X = 2

Nilai Y = 3

Nilai C = 5

Penjelasan program :

- prosedur dimulai dengan deklarasi prosedur dengan judul prosedur hitung. Variable A dan B adalah parameter formal dan integer adalah tipe parameternya.
- Variable local yang hanya dipergunakan di prosedur dan tidak termasuk parameter formal (parameter nilai), harus didefinisikan sendiri, yaitu : variable C
- Hubungan antara parameter formal di prosedur dengan parameter nyata di modul utama adalah : nilai parameter nyata X dan Y di modul utama dikirimkan ke parameter formal A dan B di prosedur. Dengan demikian nilai parameter A dan B diprosedur akan berisi nilai yang sama dengan parameter X dan Y di modul utama.

1.1.2 Pengiriman Parameter Secara Acuan

Bila pengiriman parameter secara acuan, maka perubahan-perubahan yang terjadi pada nilai parameter formal di prosedur akan mempengaruhi nilai parameter nyata. Parameter-parameter ini disebut dengan variable parameter serta dideklarasikan di deklarasi prosedur dengan menggunakan kata cadangan Var, sebagai berikut :

PROCEDURE hitung (VAR A, B, C : integer) ;

Contoh program :

Procedure Hitung (var A, B, C : integer) ;

Begin

 C := A + B ;

End ;

Var

 X, Y, Z : integer ;

Begin

 X := 2 ; Y := 3 ;

 Hitung (X, Y, Z) ;

 Writeln (' X = ', X , ' Y = ', Y, ' Z = ', Z) ;

End.

Output program :

X = 2 Y = 3 Z = 5

Penjelasan program :

- pengiriman parameter secara acuan merupakan pengiriman dua arah, bolak-balik, sehingga perubahan nilai di parameter formal akan mempengaruhi nilai parameter nyata juga. Pada contoh, nilai parameter nyata Z akan mengikuti perubahan nilai dari parameter formal C.

1.1.3 Pengiriman Parameter Sebagian Secara Nilai, Sebagian Acuan

Pengiriman parameter dapat dicampur sebagian secara nilai dan sebagian secara acuan dalam suatu prosedur. Parameter yang hanya dibutuhkan pada prosedur saja dapat dikirimkan secara nilai dan yang ingin dikirimkan balik dapat dilakukan secara acuan, sebagai berikut :

```
PROCEDURE Hitung ( A, B : integer ; Var C : integer ) ;
```

Contoh program :

```
Procedure Hitung ( A, B : integer ; Var C : integer ) ;  
Begin  
    C := A + B ;  
End ;  
Var  
    X, Y, Z : integer ;  
Begin  
    X := 2 ; Y := 3 ;  
    Hitung ( X, Y, Z ) ;  
    Writeln ( ' X = ', X , ' Y = ', Y , ' Z = ', Z ) ;  
End.
```

Output program :

X = 2 Y = 3 Z = 5

1.2 Prosedur Memanggil Prosedur Yang Lain

Prosedur dapat memanggil prosedur yang lainnya.

Contoh program :

```
Procedure Pro1 ( X1 : integer ) ;  
Begin  
    Writeln ( 'Nilai X = ', X1, 'ada di prosedur Pro1' ) ;  
End ;
```

```
Procedure Pro2 (X2 : integer ) ;  
Begin  
    Writeln ( 'Nilai X = ', X2, 'ada diprosedur Pro2' ) ;  
    Pro1 (X2) ;  
End ;
```

```
Var  
    X : integer ;  
Begin  
    X := 5 ;  
    Pro2 ( X ) ;  
End.
```

Output program :

```
Nilai X = 5 ada diprosedur Pro2  
Nilai X = 5 ada diprosedur Pro1
```

1.3 Prosedur Tersarang

Adalah prosedur yang berada didalam prosedur yang lainnya.
Bentuk Umum :

```
Program
  Procedure
    Procedure
      Begin
        ...
        ...
        ...
      End ;
    Begin
      ...
      ...
      ...
    End ;
  Begin
    ...
    ...
    ...
  end .
```

1.4 Prosedur Memanggil Dirinya Sendiri

Merupakan prosedur yang memanggil atau menggunakan prosedur itu juga. Proses dari suatu program bagian yang memanggil dirinya sendiri dikenal dengan istilah recursion. Tidak semua bahasa menyediakan kemampuan untuk melakukan proses recursion, tetapi pascal dapat. Walaupun proses ini merupakan algoritma yang baik, tetapi membutuhkan banyak memori, karena setiap kali program bagian dipanggil oleh dirinya sendiri, sejumlah ruang memori tambahan dibutuhkannya.

Contoh program :

```
Var
  I : integer ;
Procedure Rekursi ;
Begin
  Writeln ( 'Hallo saya Pascal ' ) ;
  I := I + 1 ;
  If I < 10 Then
    Rekursi ;
End ;
```

```
Begin
    I := 1 ;
    Rekursi ;
End.
```

1.5 Acuan Forward

Digunakan untuk mendeklarasikan dimuka judul prosedur terpisah dari bloknnya. Judul prosedur yang berisi parameter terpisah ini merupakan judul yang semestinya dan di deklarasikan dengan menambahkan kata cadangan “ Forward ”. Blok prosedur ini letaknya terpisah dari judulnya, juga diawali dengan judul prosedur, tetapi hanya mencantumkan judul atau nama prosedurnya saja, tanpa mencantumkan parameter formalnya.

Contoh program :

```
Procedure Pro1 (Var I : integer ) ; Forward ;
Procedure Pro2 (Var I : integer ) ;
Begin
    Writeln ( ‘ Prosedur PRO’, I ) ;
End ;
```

```
Procedure Pro1 ;
Begin
    Writeln ( ‘Prosedur PRO’, I ) ;
End ;
```

```
Var
    I : integer ;
Begin
    I := 1 ;
    Pro1 ( I ) ;
    I := 2 ;
    Pro2 ( I ) ;
End.
```

Output :

```
Prosedur PRO1
Prosedur PRO2
```

1.6 Prosedur Standar

1.6.1 Prosedur Standar Exit

Digunakan untuk keluar dari suatu blok. Bila diletakkan pada program bagian, prosedur standar exit akan menyebabkan proses di program bagian berhenti dan proses kembali ke blok program yang memanggilnya. Bila berada di program utama, menyebabkan proses program berhenti.

1.6.2 Prosedur Standar Halt

Digunakan untuk menghentikan proses program baik di program bagian maupun program utama.

1.6.3 Prosedur Standar Move

Bentuk umum :

Move (var source, dest; count : word)

Digunakan untuk menyalinkan suatu blok sebanyak count byte memori dari blok dimulai byte pertama source dan disalinkan ke byte pertama dest.

Contoh program :

```
Var
    Sumber, Tujuan : string [5] ;
    Sejumlah : word ;
Begin
    Sumber := 'ABCDE' ;
    Tujuan := 'FGHIJ' ;
    Sejumlah := 4 ;
    Move (Sumber, Tujuan, Sejumlah) ;
    Writeln (Sumber : 7, Tujuan : 7) ;
End.
```

Output program :

ABCDE ABCIJ

1.6.4 Prosedur Standar FillChar

Digunakan untuk mengisi sejumlah byte nilai ke dalam suatu variable, sbb :

FillChar (x ; count : word ; ch)

X adalah variable yang dapat bertipe apapun yang akan diisi dengan nilai tipe ordinal ch sebanyak count byte.