

JENIS DATA SEDERHANA & INPUT/OUTPUT DATA

1.1 Tipe Data

A. Data Tipe Sederhana

Dihubungkan dengan sebuah identifier untuk sebuah data. Data tipe ini digolongkan menjadi tipe data standar dan tipe data yang didefinisikan oleh pemakai.

- Tipe data standar :
Integer, real, char, string, Boolean
- Tipe data yang didefinisikan pemakai
Enumerated atau scalar type, subrange type

B. Data Tipe Terstruktur

Terdiri dari beberapa data item yang dihubungkan satu dengan lainnya. Masing-masing grup dari data item dihubungkan dengan suatu identifier tertentu. Ada 4 macam yang termasuk dalam data ini : array, record, file dan set

C. Data Tipe Penunjuk

Digunakan untuk membuat data terstruktur tipe dinamik.

1.1.1 Integer

Merupakan nilai bilangan bulat baik dalam bentuk desimal maupun hexadecimal. Ada 5 macam tipe data integer berdasarkan jangkauan nilai, yaitu :

Tipe	Ukuran Memori (dalam byte)	Jangkauan Nilai
Byte	1	0 ... 255
Shortint	1	- 128 ... 127
Integer	2	- 32768 ... 32767
Word	2	0 ... 65535
Longint	4	- 2147483648 ... 2147483647

Contoh (1) :

Tipe data byte digunakan jika nilai tidak lebih dari 255

```
Var
    Jumlah : byte ;
Begin
    Jumlah := 200 ;
    Writeln (' Nilai jumlah = ', Jumlah);
End.
```

Contoh (2) :

Bila nilai yang digunakan lebih besar dari 255, tetapi tidak lebih besar dari 32767, maka dapat digunakan tipe integer

```
Var
    Jumlah ; integer ;
Begin
    Jumlah := 25000 ;
    Writeln (' nilai jumlah = ', Jumlah) ;
End.
```

1.1.2 Real

Nilai konstanta numeric real berkisar dari 1E-38 sampai 1E+38. E menunjukkan nilai 10 pangkat, dan tipe data ini menempati memori sebesar 6 byte.

Contoh :

```
Var
    Nilai1, Nilai2 : real ;
Begin
    Nilai1 := 12345678901.2345 ;
    Nilai2 := 12345 ;
    Writeln (' Nilai 1 = ', Nilai1) ;
    Writeln (' Nilai 2 = ', Nilai2) ;
End.
```

Output dari contoh program diatas :

```
Nilai1 = 1.2345678901E+10
Nilai2 = 1.2345000000E+04
```

1.1.3 Karakter

Nilai data karakter berupa sebuah karakter yang ditulis diantara tanda petik tunggal, seperti : ' A '. penggunaan variable untuk menyimpan tipe data karakter ini harus dideklarasikan dengan tipe Char.

Contoh :

```
Var
    Huruf : char ;
Begin
    Huruf := ' D ' ;
    Writeln ( ' Hurufnya adalah : ', Huruf ) ;
End.
```

Output program :

Hurufnya adalah : D

1.1.4 String

Merupakan urutan dari karakter yang terletak diantara tanda petik tunggal. Nilai data string akan menempati memori sebesar banyaknya karakter string ditambah dengan 1byte. Bila panjang dari suatu string didalam deklarasi variable tidak disebutkan, maka dianggap panjangnya adalah 255 karakter.

Contoh :

```
Var
    Tanggal    : string [10] ;
    Hari       : string [6] ;
    Jarak      : char ;
Begin
    Tanggal := ' 19 Februari 2007 ' ;
    Hari := ' Senin ' ;
    Jarak := ' ' ;
    Writeln (Tanggal, Jarak, Hari) ;
End.
```

1.1.5 Boolean

Mempunyai dua buah nilai, yaitu true dan false.

Contoh :

Var

Lagi : Boolean ;

Begin

Lagi := True ;

Writeln ('nilai lagi adalah ', Lagi) ;

End.

1.2 Binary Operator

Disebut dengan binary operator karena operator ini digunakan untuk mengoperasikan dua buah operand. Operand dapat berbentuk konstanta ataupun variable. Digunakan untuk operasi aritmatika yang berhubungan dengan tipe data integer dan real.

Operator	Operasi	Tipe Operand	Tipe Hasil
*	Perkalian	Real, real Integer, integer Real, integer	Real Integer Real
DIV	Pembagian bulat	Integer, integer	Integer
/	Pembagian real	Real, real Integer, integer Real, integer	Real Real Real
MOD	Modulus (sisa pembagian)	Integer, integer	Integer
+	Pertambahan	Real, real Integer, integer Real, integer	Real Integer Real
-	Pengurangan	Real, real Integer, integer Real, integer	Real Integer real

1.3 Relational Operator

operator hubungan digunakan untuk membandingkan hubungan antara dua buah operand dan akan didapatkan hasil tipe Boolean, yaitu, true atau false.

Operator	Operasi
=	Sama dengan
< >	Tidak sama dengan
>	Lebih besar dari
> =	Lebih besar sama dengan dari
<	Lebih kecil dari
< =	Lebih kecil sama dengan dari
IN	Seleksi dari anggota himpunan

1.4 Input Dan Output

1.4.1 Memasukkan Data

Cara yang paling banyak digunakan untuk memasukkan data adalah dengan mengetikkannya melalui keyboard. Prosedur yang digunakan untuk melakukannya dalam pascal adalah *Read* dan *Readln*.

Perbedaan perintah *read* dan *readln* adalah :

Readln : untuk memasukkan data per baris, setelah menekan tombol enter, maka akan ganti baris

Read : untuk memasukkan data dan tidak berganti baris, masih dalam baris yang sama

Contoh :

Var

Sisi, panjang, luas : integer ;

Begin

Write ('Masukkan sisi persegi panjang =');

Readln (sisi) ;

Write ('Masukkan panjang persegi panjang =');

Readln (panjang) ;

Luas := sisi x panjang ;

Writeln ('Luas persegi panjang adalah', Luas) ;

End.

Output program :

Masukkan sisi persegi panjang = 10 → data yg diinput melalui keyboard

Masukkan panjang persegi panjang = 10

Luas persegi panjang adalah 100

1.4.2 Menampilkan Hasil

untuk menampilkan hasil digunakan prosedur standar *Write* atau *Writeln*.

Perbedaan perintah *write* dan *writeln* adalah :

Write : menampilkan hasil tanpa ganti baris dan tampilan berikutnya akan disambung dalam baris yang sama

Writeln : untuk menampilkan tampilan per baris, akan ganti baris untuk tampilan berikutnya.

(untuk contoh program, dapat dilihat contoh di sub bab 1.4.1 Memasukkan Data)

