

## BAB 3

### INPUT PROSES OUTPUT

#### **Target Pencapaian**

Siswa mengerti Penggunaan Statement Input, Proses dan Output dalam pembuatan Program dengan Pascall

### 3.1. Input Output

Setiap program aplikasi pasti membutuhkan operasi input dan output. Memasukan data input secara interaktif merupakan salah satu contoh dari program aplikasi yang banyak beredar saat ini. Suatu program yang tidak dapat menghasilkan bentuk keluaran (*output*) akan menjadi sebuah program yang sia-sia, karena kita tidak dapat melihat hasil dari proses yang dilakukan oleh program tersebut.

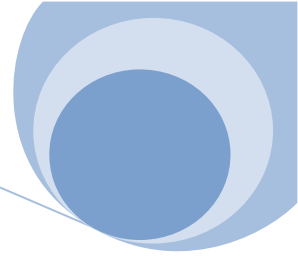
Contoh Input, Proses Output;

Contoh 3.1

```
1. Uses Crt;
2. Var
3. Bil1, Bil2, Bil3 : Integer;
4. Begin
5.     { * Bagian Input * }
6.     Write('Masukan Bilangan ke satu : ');Readln(Bil1);
7.     Write('Masukan Bilangan ke dua : ');Readln(Bil2);
8.
9.     { * Bagian Proses * }
10.    Bil3 := Bil1+Bil2;
11.
12.    { * Bagian Output * }
13.    Writeln('Bilangan 3 = ',Bil3);
14. End.
```

### 3.2. Memasukan Data

Dalam Turbo Pascal telah disediakan prosedur standar untuk membaca data dari keyboard, prosedur tersebut adalah prosedur standar *Read* dan *Readln*. Prosedur standar *Read* dan *Readln* mempunyai aturan tertentu untuk beberapa tipe pengenalan variabel. Tiap-tiap data yang dimasukan harus sesuai dengan tipe variabelnya.



#### Contoh 3.2

```
1. Var
2. A, B : Integer
3. Begin
4.   Write('Masukan Nilai A:');Readln(A);
5.   Write('Masukan Nilai B:');Readln(B);
6. End.
```

### 3.3. Menampilkan Hasil

Dalam Turbo Pascal telah disediakan prosedur standar untuk menampilkan hasil ke layar monitor, yaitu prosedur standar *Write* dan *Writeln*. Prosedur standar *Write* berguna untuk menampilkan hasil tanpa ganti baris, sedangkan dengan *Writeln* setelah menampilkan hasil akan langsung menuju baris berikutnya. Prosedur ini sudah sering Anda gunakan dalam contoh program di atas.

#### Contoh 3.3

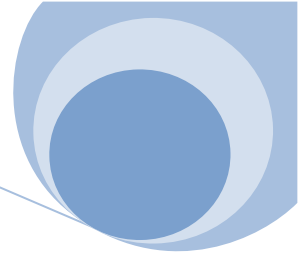
```
1. Var
2. A, B : Integer
3. Begin
4.   A := 10;
5.   B := 3;
6.   Writeln('Isi Nilai A:',A);
7.   Writeln('Isi Nilai B:',B);
8. End.
```

### 3.4. Mengatur letak di Layar

Dalam Turbo Pascal telah disediakan prosedur standar untuk menampilkan tulisan atau data pada posisi tertentu dari layar. Prosedur tersebut adalah prosedur *GotoXY*.

Prosedur standar ini digunakan untuk meletakkan cursor di posisi layar yang ditunjukkan oleh nilai XPOS dan YPOS.

Nilai XPOS adalah nilai untuk sumbu Horizontal sedangkan nilai YPOS adalah untuk sumbu Vertikal.



#### Contoh 3.4

```
1. Uses Crt;  
2. Begin  
3.   Clrscr;  
4.   Gotoxy(10,15);Writeln('TAK SELAMANYA CINTA ITU INDAH');  
5. End.
```

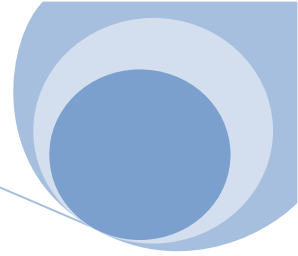
### 3.5. Menyisipkan dengan INSLINE

Dalam Turbo Pascal telah disediakan prosedur standar INSLINE, Prosedur ini digunakan untuk menyisipkan sebuah baris pada posisi kursor dan menggeser kebawah tampilan-tampilan baris dibawahnya.

#### Contoh 3.5

```
1. Uses Crt;  
2. Var  
3. I : Integer;  
4. Tulisan : String;  
5. Begin  
6.   Clrscr;  
7.   Tulisan := 'TEKNIK INFORMATIKA';  
8.   For I := 1 To 25 Do  
9.     Begin  
10.      Writeln(Tulisan);  
11.      Delay(100);  
12.    END;  
13.   Readln;  
14.   For I := 1 To 25 Do  
15.     Begin  
16.      Gotoxy(1,1);Inslne;  
17.      Delay(100);  
18.    End;  
19. End.
```

Program ini akan menyebabkan tulisan yang ditampilkan menghilang perbaris ke bawah layar.



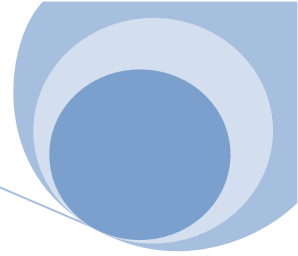
### 3.6. Menghapus dengan DELLINE

Dalam Turbo Pascal telah disediakan prosedur standar DELLINE, Prosedur ini digunakan untuk menghapus sebuah baris pada posisi kursor dan menggeser ke atas tampilan-tampilan baris dibawahnya.



Contoh 3.6

```
1. Uses Crt;  
2. Var  
3. I : Integer;  
4. Tulisan : String;  
5. Begin  
6. Clrscr;  
7. Tulisan := 'PEMROGRAMAN DENGAN PASCAL';  
8. gotoxy(1,1);Writeln(tulisan);  
9. readln;  
10. gotoxy(1,1);  
11. DELLINE;  
12. Delay(100);  
13. Readln;  
14. End.
```



### 3.7. Mengatur warna tampilan

Dalam Turbo Pascal telah disediakan prosedur standar yang dapat digunakan untuk mengatur warna tampilan teks di layar, yaitu *Textcolor*, *Textbackground*, *LowVideo*, *NormVideo* dan *HighVideo*.

#### 3.7.1. Mengatur Warna Depan (*foreground*)

Untuk mengatur warna foreground dari karakter-karakter di layar dapat dipergunakan prosedur standar *textColor* Sebagai berikut:

**Textcolor(Color : Byte);**

Penggunaan prosedur ini harus melibatkan unit standar Crt dan di dalam unit standar ini telah didefinisikan beberapa konstanta yaitu :

Penggunaan prosedur ini harus melibatkan unit standar Crt dan di dalam Unit standar ini telah didefinisikan beberapa konstanta, yaitu :

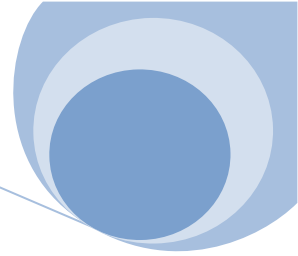
WARNA	KODE	WARNA	KODE
Black	0	LightBlue	9
Blue	1	LightGreen	10
Green	2	LightCyan	11
Cyan	3	LightRed	12
Red	4	LightMagenta	13
Magenta	5	Yellow	14
Brown	6	White	15
LightGray	7	Blink	128
DarkGrey	8		

#### 3.7.2. Mengatur Warna Latar Belakang

Untuk mengatur warna background dari karakter-karakter dilayar, dapat dipergunakan prosedur standar ***Textbackground*** sebagai berikut :

**TextBackground(Color:Byte);**

Sama seperti *Textcolor*, penggunaan prosedur ini harus melibatkan Unit CRT dan didalam unit standar ini telah didefinisikan beberapa konstanta, yaitu :



WARNA	KODE
Black	0
Blue	1
Green	2
Cyan	3
Red	4
magenta	5
Brown	6
LightGray	7

### Contoh 3.7

```

1. Uses Crt;
2. Begin
3.   Clrscr;
4.   Textcolor(White);Textbackground(Red);
5.   Writeln('Warna Tulisan Putih Latar Belakang Merah');
6.   Textcolor(Magenta);Textbackground(Green);
7.   Writeln('Warna Tulisan Magenta Latar Belakang Hijau');
8.   Textcolor(14);Textbackground(4);
9.   Writeln('Warna Tulisan Kuning Latar Belakang Merah');
10.  Textcolor(10+Blink);Textbackground(9);
11.  Writeln('Warna Tulisan Hijau Terang Latar Belakang Biru Terang Berkedip');
12.  Readln;
13. End.

```

## 3.8. Membuat Jendela

Dengan menggunakan prosedur standar **Window**, maka dimungkinkan untuk membuat suatu jendela (window) yang letaknya dimanapun dalam layar. Jika anda membuat suatu window, maka jendela tersebut akan berperan seperti layar yang utuh dengan mengabaikan bagian layar lainnya diluar jendela.

Sintak :

### Window(X1,Y1,X2,Y2 : Byte)

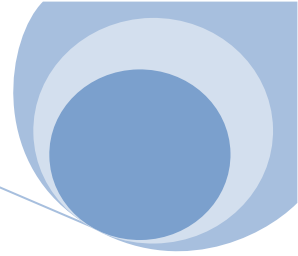
Dengan X1 dan Y1 adalah koordinat ujung paling kiri atas dan X2, Y2 adalah koordinat ujung kanan bawah layar.

#### Contoh 3.7.

```
1. Uses Crt;  
2. Begin  
3.   Window(1,1,80,25);  
4.   Textcolor(1);Textbackground(1);  
5.   Clrscr;  
6.   Window(10,4,70,10);  
7.   Textcolor(14);Textbackground(4);  
8.   Clrscr;  
9.   Writeln('Jendela Ke Satu ');  
10.  Window(12,7,50,20);  
11.  Textcolor(13);Textbackground(2);  
12.  clrscr;  
13.  Writeln('Jendela Ke Dua');  
14.  Window(2,10,60,18);  
15.  Textcolor(15);Textbackground(3);  
16.  clrscr;  
17.  Writeln('Jendela Ke Tiga');  
18.  Readln;  
19. End.
```

Contoh Output yang dihasilkan :





## Latihan

---

3.1. Diketahui sebuah kotak mempunyai ukuran

Panjang Alas = 10 cm

Lebar Alas = 7 cm

Tinggi = 15 cm

Buatlah program untuk menghitung isi kotak dan luas permukaan kotak. Tampilan yang diinginkan adalah seperti berikut:

Ukuran Kotak = 10 cm \* 7 cm \* 15 cm

Isi kotak = 1050 cm<sup>3</sup>

3.2. Perusahaan Telepon di kota 'X' mempunyai tarif Rp. 150 per-pulsa. Setiap langganan dikenakan biaya langganan Rp. 25000,- per-bulan.

Buatlah program untuk menghitung jumlah tagihan kepada salah seorang pelanggan

Masukan : Nama Langganan

Jumlah Pulsa

Keluaran: Nama Langganan      Jumlah Tagihan

3.3. Soal sama dengan nomor 2, ubahlah bentuk layar masukan dan keluaran seperti berikut:

Nomor Langganan :

Nama :

Alamat :

Perincian Tagihan

Biaya Langganan : Rp.....

Pulsa : Rp. .... \* JumlahPulsa

Total : Rp.....