

MODUL 1

Struktur Bahasa PASCAL secara umum

Pascal mempunyai struktur sebagai berikut:

1. Bagian Judul Program
2. Bagian Deklarasi e
 - a. Deklarasi tipe data (TYPE)
 - b. Deklarasi variabel (VAR)
 - c. Deklarasi konstanta (CONST)
 - d. Deklarasi label (LABEL)
 - e. Deklarasi sub-program (PROCEDURE dan FUNCTION)
3. Bagian Program Utama Perintah-perintah.

Teks Pascal setidaknya memiliki bagian Judul Program, bagian Deklarasi, dan Bagian Program Utama yang berupa perintah-perintah. Sedangkan untuk bagian deklarasi menyesuaikan dengan isi dari program itu sendiri. Contoh program PASCAL:

```
program TAMBAH_00; { Menjumlahkan dua bilangan yang nilainya diberikan
dalam perintah}
var X, Y, Z: integer; { Deklarasi variabel X,Y dan Z sebagai bilangan bulat }
BEGIN { Program Utama Mulai }
X := 50; { Perintah memberikan nilai 50 pada var. X }
Y := 25; { Perintah memberikan nilai 25 pada var. Y }
Z := X + Y; { Perintah menjumlahkan X dan Y serta menyimpan hasilnya ke Z}
END. { Akhir Program Utama }
```

Pada contoh ini nilai X dan Y tidak bisa sembarang, karena didefinisikan tertentu.

Agar nilai X dan Y bisa bebas ditentukan, nilai X dan Y dibaca dari default input.

program TAMBAH_01; { Menjumlahkan dua buah bilangan yang dibaca dari default input }
var X, Y, Z: integer; { Deklarasi variabel X,Y dan Z sebagai bilangan bulat }
BEGIN { Program Utama Mulai }
read(X); { Membaca nilai X lewat key-board }
read(Y); { Membaca nilai Y lewat key-board }
Z := X + Y; { Menjumlahkan X dan Y serta menyimpan hasilnya ke Z }
write(Z); { Menyajikan Z ke layar monitor }
END. { Akhir Program Utama }

Dasar Bahasa PASCAL

Unsur-unsur Pemrograman

- Mendapatkan data dengan membaca data dari default input (key board, file atau sumber data lainnya).
- Menyimpan data ke dalam memori dengan struktur data yang sesuai,
- Memproses data dengan instruksi yang tepat.
- Menyajikan atau mengirimkan hasil olahan data ke default output (monitor, file atau tujuan lainnya).

Jenis identifier

a. Identifier umum

Merupakan identifier yang didefinisikan sendiri oleh pemrogram. Pemrogram mempunyai kebebasan untuk menentukan nama identifiernya, dengan syarat nama tersebut tidak sama dengan identifier standar dan reserved word yang akan dibahas lebih lanjut. Hal ini untuk mencegah kesalahan yang bisa timbul akibat tumpang tindih identifier dalam program.

b. Identifier Standar (Baku)

Merupakan identifier yang didefinisikan oleh pembuat kompilasi Pascal. Biasanya pembuat kompilasi menyediakan suatu library yang sudah ada didalam kompilasi. Library berisi berbagai procedure, fungsi atau unit yang sudah siap pakai. Misalnya Turbo Pascal Windows 1.5 memiliki suatu unit untuk memproses output yaitu wincrt, gotoxy, yang dengan mudah bisa dipakai oleh programmer di dalam menuliskan kode-kode programnya. Dinamai Identifier Standar karena suatu kompilasi tidak harus memilikinya, masing-masing kompilasi dimungkinkan

mempunyai identifier yang berbeda untuk suatu tugas yang hampir sama. Misalnya Turbo Pascal versi DOS menggunakan crt untuk melakukan fungsi yang sama dengan winCRT (TPW 1.5). Beberapa Identifier Standar yang dimiliki oleh kompiler-kompiler Pascal antara lain:

abs arctan boolean char cos dispose eof eoln exp false input integer ln maxint
new odd ord output pack page pred read readln real reset rewrite round sin sqr
sqrt succ text true trunc write writeln

c. Identifier "reserved word", yaitu yang sudah didefinisikan dan digunakan oleh bahasa PASCAL sendiri (Kita tidak bisa menamai identifier kita dengan ini).

and array begin case const div do downto else end file for forward function goto
if in label mod nil not of or packed procedure program record repeat set then to
type until var while with

Deklarasi Variable:

Mendeklarasikan variabel adalah:

- Memberikan nama variabel sebagai identitas pengenalan
- Menentukan tipe data variabel

Contoh deklarasi variabel:

var K : integer;
R : real;
C : char;
T : boolean;

Beberapa identifier yang sejenis bisa dideklarasikan bersamaan.

var i, j, k : integer; {Variabel i, j dan k sebagai integer}
namaMHS, alamatMHS : char; {Nama dan alamat mahasiswa }

Deklarasi Konstanta:

Mendeklarasikan konstanta adalah: a. Memberikan nama konstanta sebagai identitas pengenalan

b. Menentukan nilai konstanta

Contoh deklarasi konstanta:

```
const MaximumSize = 100; {integer }  
ExitCommand = 'Q'; {char }
```

Tipe Data

Tipe data yang disediakan oleh PASCAL meliputi:

1. Tipe Data Sederhana

merupakan tipe data dasar yang sering dipakai oleh program, meliputi: integer (bilangan bulat), real (bilangan pecahan), char (alphanumeric dan tanda baca), dan boolean (logika). Untuk data integer dan real masing-masing terbagi menjadi beberapa kategori

a. Bilangan Integer

merupakan tipe data berupa bilangan bulat, terbagi atas beberapa kategori seperti terlihat dalam tabel 1. tabel 1 menunjukkan jenis data, ukuran dalam memori dan rentang nilainya.

tabel 1. Tipe Data Bilangan Integer

Tipe Data	Ukuran Tempat	Rentang Nilai
Byte	1 byte	0 s/d +255
Shortint	1 byte	-28 s/d +127
integer	2 bytes	-32768 s/d 32767
Word	2 bytes	0 s/d 65535
Longint	4 bytes	2147483648 s/d 2147483647

Contoh bilangan integer adalah: 34 6458 -90 0 1112 Penggolongan tipe data integer tersebut dimaksudkan untuk membatasi alokasi memori yang dibutuhkan misalkan untuk suatu perhitungan dari suatu variabel bilangan diperkirakan nilai maksimumnya 32767 kita cukup mendeklarasikan variabel

bilangan sebagai integer (2 byte), daripada sebagai longint(4 byte). Di dalam kompilernya, Pascal menyediakan konstanta untuk bilangan Integer yaitu: MaxInt and MaxLongInt, pemrogram bisa menggunakannya di dalam programnya tanpa harus terlebih dahulu mendefinisikannya.

-MaxInt bernilai 32.767

-MaxLongint bernilai 2.147.483.647.

contoh:

```

Program display_maxint;
uses wincrt;
begin writeln (maxint)
end.

```

Hasilnya: 32.767

b. Bilangan Real

Bilangan real atau nyata merupakan jenis bilangan pecahan, dapat dituliskan secara biasa atau model scientific . Contoh bilangan real: 34.265 -3.55 0.0 35.997E+11, dimana E merupakan simbol perpangkatan 10. Jadi 452.13 mempunyai nilai sama dengan 4.5213e2. Penggolongan tipe data bilangan real dapat dilihat pada tabel 2. Bilangan Real

Tabel 2. Tipe Data Bilangan Real

Tipe Data	Ukuran Tempat	Rentang Nilai
real	6 bytes	2.9 x 10 ⁻³⁹ s/d 1.7 x 10 ³⁸
single	4 bytes	1.5 x 10 ⁴⁵ s/d 3.4 x 10 ³⁸
double	8 bytes	5.0 x 10 ⁻³²⁴ s/d 1.7 x 10 ³⁰⁸
extended	10 bytes	3.4 x 10 ⁻⁴⁹³² s/d 1.1 x 10 ⁴⁹³²
comp	8 bytes	-9.2x 10 ¹⁸ s/d 9.2x 10 ¹⁸

c. Char

Tipe data ini menyimpan karakter yang diketikkan dari keyboard, memiliki 266 macam yang terdapat dalam tabel ASCII (American Standard Code for Information Interchange). Contoh: 'a' 'B' '+', dsb. Yang perlu diingat bahwa dalam menuliskannya harus dengan memakai tanda kutip tunggal. Jenis data ini memerlukan alokasi memori sebesar 1(satu) byte untuk masing-masing data.

d. Tipe Data Boolean

merupakan tipe data logika, yang berisi dua kemungkinan nilai: TRUE (benar) atau FALSE (salah). Turbo Pascal for Windows memiliki tiga macam jenis ini yaitu: Boolean, WordBool, dan LongBool. Tipe boolean memakai memori paling kecil, sedangkan WordBool dan LongBool dipakai untuk menulis program yang sesuai dengan lingkungan Windows.

Tabel 3. Tipe Data Boolean

Tipe Data	Ukuran
	Tempat
Boolean	1 byte
WordBool	2 byte
Longbool	3 byte

Sebagai bilangan ordinal boolean true mempunyai nilai 1(satu), sedangkan false nilainya adalah 0(nol).

```
Contoh: Program display_bool;  
uses wincrt; begin  
writeln(ord(true));  
writeln(ord(false)); end.
```

Hasilnya: 1 0 3.2.

Tipe Data Terstruktur

tipe ini terdiri atas : array, record, set, dan file. String adalah tipe data jenis array, tetapi karena string memiliki kekhasan tersendiri sebagai array dari karakter

maka penulis perlu memberikan penjelasan tersendiri. Sedangkan untuk array, record, dan file perlu dijelaskan dalam bab yang lain karena agak banyak hal-hal yang perlu dibahas.

e. Tipe Data String

merupakan suatu data yang menyimpan array (larik), sebagai contoh 'ABCDEF' merupakan sebuah konstanta string yang berisikan 6 byte karakter. Ukuran Tempat untuk tipe data ini adalah 2 s/d 256 byte, dengan jumlah elemen 1 s/d 255. String dideklarasikan dengan string [konstanta] atau string. Bila ukuran string tidak didefinisikan maka akan banyak memakan ruang, karena ukuran string menyesuaikan dengan defaultnya. Misalkan var kata: string [20]; atau var kata: string; karena string merupakan array dari karakter. Maka kata[1] merupakan karakter pertama dari string, kemudian kata[2], merupakan elemen kedua, dst.

Contoh:

Program hal_string;
Uses wincrt;
var s : string;
begin s:='Hello';
writeln(s);
writeln('panjang dari string adalah: ',ord(s[0]));
end.
Karakter nol merupakan karakter yang menyatakan panjang string. Sehingga ord(s[0]) menyatakan panjang dari string tersebut. Panjang string juga bisa dinyatakan sebagai length(s).

MODUL 2 ARRAY (LARIK)

1. Pendahuluan

Suatu array adalah sebuah struktur data yang terdiri atas banyak variabel dengan tipe data sama, dimana masing-masing elemen variabel mempunyai nilai indeks. Setiap elemen array mampu untuk menyimpan satu jenis data (yaitu: variabel).

Suatu array dinyatakan dengan type, sehingga variabel yang bekerja akan dinyatakan dengan: contoh type $A = \text{array } [1..10] \text{ of integer};$

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Secara logika pendefinisian array di atas merupakan sekumpulan kotak ,
dimana tiap kotak mempunyai nilai indeks integer 1, 2, 3, ...,9, 10 tiap elemen
array ditandai dengan: $A[1], A[2], A[3], A[4], A[5], A[6], A[7], A[8], A[9],$
 $A[10]$

2. Sifat Array

Array merupakan struktur data yang statis, yaitu jumlah elemen yang ada harus ditentukan terlebih dahulu dan tak bisa di ubah saat program berjalan. Untuk menyatakan array dalam PASCAL kita harus terlebih dahulu: Mendefinisikan jumlah elemen array,–

Mendefinisikan– tipe data dari elemen array

Contoh. $\text{const } N=10; \text{ type}$
 $A = \text{array } [1..N] \text{ of integer};$

3. Array Satu Dimensi

3. Array Satu Dimensi

Pernyataan di atas merupakan penjelasan tentang array dengan satu dimensi.

Pendefinisian array secara umum adalah sebagai berikut: jika kita ingin membuat beberapa array dengan tipe/jenis yang sama, kita lebih baik jika mendeklarasikan dengan tipe selanjutnya dengan deklarasi var.

SYNTAX

```
Type nama_array = ARRAY[bawah..atas] of tipe_data;  
var variabel_array : nama_array;
```

atau dengan menggunakan statemen var :

```
var variabel_array : ARRAY[bawah..atas] of tipe_data;
```

Penjelasan: Bawah dan Atas menyatakan batas untuk array. tipe_data adalah merupakan tipe variabel yang dipunyai array (mis. Integer, char, real, dsb)
Contoh: type intarray = ARRAY [1..20] of integer; Pernyataan diatas adalah pernyataan untuk membentuk suatu array bernama intarray, yang berisi 20 tempat untuk bilangan integer. Setiap posisi disebut elemen, yang menyimpan suatu bilangan integer. langkah berikutnya adalah membuat suatu variabel kerja dengan tipe intarray yaitu, var numbers : intarray; kita bisa melakukan operasi pada setiap elemen dari numbers secara individual.

Contoh kita bisa memberi nilai pada suatu elemen array seperti berikut:

```
numbers[2] := 10;
```

perintah ini memberikan suatu nilai integer 10 pada elemen ke-2 dari array numbers. Nomor dari elemen ditempatkan didalam kurung tegak. Contoh berikut adalah merupakan array yang menyimpan variabel-variabel integer. Data dengan

tipe integer hanya bisa dimasukkan satu persatu, kemudian baru bisa ditampilkan di monitor secara bersamaan

Contoh a.

```
program INT_ARRAY;
uses wincrt;
const N=10;
type int_array = ARRAY [1..N] of integer;
var bil : int_array;
indeks : integer;
BEGIN
writeln('masukkan sepuluh bilangan integer. ');
for indeks := 1 to 10 do
begin
readln(bil[indeks]); { loop untuk memasukkan elemen array }
end;
writeln('Isi dari array ini adalah'); { tampilkan setiap elemen }
for indeks := 1 to 10 do
begin
writeln('bil[' , indeks:2, ' ] adalah ', bil[indeks] );
end
END.
```

Contoh b.

```
program contoh_ARRAY;
uses wincrt;
var
a : array[1..10] of byte; {maksimum jumlah elemen=10}
begin
a[1]:=10;
a[2]:=15;
a[3]:=a[1]+a[2];
writeln(a[1]);
writeln(a[2]);
writeln(a[3]);
end.
```

Latihan

1. Menggunakan Array 1 dimensi buatlah program dengan ketentuan:

Input---→ Nilai PPN, Nilai PPA, Nilai Logika, Nilai Agama

Output-→ Total Nilai

Ket: Nama Array = nilai

Nama variabel = n

Jumlah Range = 5

2. Untuk soal no.1 tambahkan proses untuk mendapatkan kelulusan jika nilai logika > 7 dan proses untuk mendapatkan grade (A jika total >34, B jika total > 28, C jika total > 24 dan D jika total <=24).