

Array dan record

Materi Struktur Data Pascal

Pengertian

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).

Cara Pendefinisian

- contoh type
A = array [1..6] of integer;

1	2	3	4	5	6
---	---	---	---	---	---

- Secara logika pendefinisian array di atas merupakan sekumpulan kotak , dimana tiap kotak mempunyai nilai indeks integer 1, 2, 3, ...,6
tiap elemen array ditandai dengan:
A[1], A[2], A[3], A[4], A[5], A[6]
untuk mengisi elemen array misal A[1]:=4;.

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

Contoh. const N=10;
type
A= array [1..N] of integer;

Array 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 type selanjutnya dengan deklarasi var.

Sintax array 1 dimensi

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

Contoh Program Array 1 dimensi

```
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.
```

Array Multidimensi

- Dalam array multidimensi terdiri atas baris (row) dan kolom (column). Index pertama adalah baris dan yang kedua adalah kolom

Syntax

- Type nama_array =ARRAY[bawah..atas, bawah..atas] of tipe_data;
var variabel_array : nama_array; **atau**
- SYNTAX var variabel_array :
ARRAY[bawah..atas, bawah..atas] of tipe_data;
- Pernyataan berikut membentuk suatu array integer dengan nama bilangan , 10 x 10 elemen (100).
type matriks = ARRAY [1..10, 1..10] of integer;
var AKU: matriks;

Array[1..5,1..5]

A[1,1]	A[2,1]			A[5,1]
A[2,1]	A[2,2]			
		A[3,3]		
			A[4,4]	
A[5,1]				A[5,5]

Contoh Program

```
Procedure ISI_MATRIK(AKU:matriks;  
m,n:integer);  
var  
i,j: integer; {faktor pengulang}  
begin  
for i:=1 to m do  
begin  
for j:=1 to n do  
begin  
gotoxy(x+2i.y+2y);  
read(A[i,j]);  
end;  
readln ;{ini memungkinkan kita menulis tiap  
baris elemen}  
end;  
end;
```

Record

- Sebuah record rekaman disusun oleh beberapa field. Tiap field berisi data dari tipe dasar / bentukan tertentu. Record mempunyai kelebihan untuk menyimpan suatu sekumpulan elemen data yang berbeda-beda tipenya (di banding array). Contoh , sebuah record dengan empat buah field

Cara Pendefinisian

- Cara pendeklarasian dari record adalah sbb:

field1	field2	field3	field4

- Mendefinisikan tipe dari record (jumlah field, jenis tipe data yang dipakai),
- Mendefinisikan variabel untuk dilakukan operasi.

Sintax

```
type
nama_record = record
  identifier_1 : tipe_data_1;
  :
  :
  identifier_n : tipe_data_n;
end;
var variabel : nama_record;
```

Contoh

```
type  
Data_mahasiswa = record  
  Nama : string;  
  Usia : integer;  
  Kota : String;  
  Kodepos : integer;  
end;  
Var  
x: Data_mahasiswa;
```

Pengaksesan Elemen Record

```
x.Nama  
x.Usia  
x.Kota  
x.Kodepos
```

Contoh Program

```
program RECORD_INTRO;  
type tanggal = record  
  bulan, hari, tahun : integer;  
end;  
var waktu : tanggal;  
begin  
  waktu.hari :=25;  
  waktu.bulan:=09;  
  waktu.tahun:= 1983;  
  writeln('hari ini adalah  
' ,waktu.hari,',',waktu.bulan,',', waktu.tahun)  
end.
```

Penggunaan With ... do

Pernyataan with untuk lebih menyederhanakan pengaksesan field-field pada record. Pemrograman dapat mengakses field cukup dengan menyebutkan nama field-nya saja.

- x.Nama
x.Usia
x.Kota
x.Kodepos diganti
- with x do
Begin
Nama
Usia
Kota
Kodepos
end

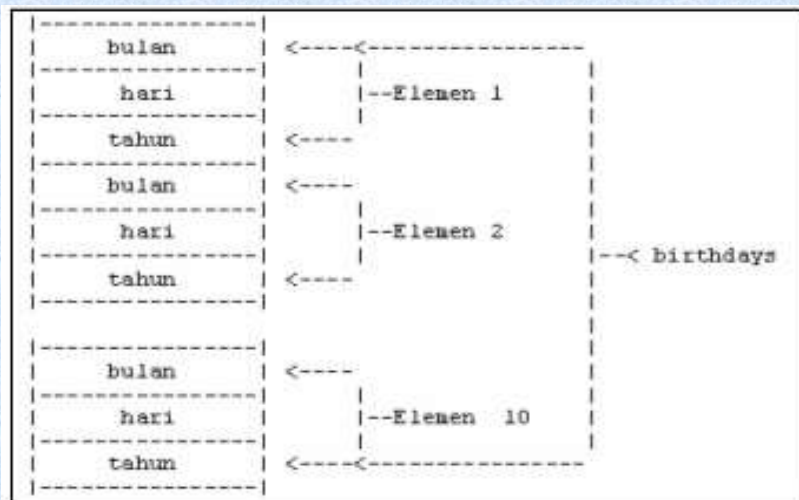
Contoh

```
program RECORD_INTRO;  
  type tanggal = record  
    bulan, hari, tahun : integer;  
  end;  
  var waktu : tanggal;  
  begin {program utama}  
    with waktu do {mulai with}  
      begin  
        hari :=25;  
        bulan:=09;  
        tahun:=1983;  
        writeln('hari ini adalah ',hari,':',bulan,':', tahun)  
      end {akhir with}  
    end.
```

Array dari Record

```
type tanggal = record  
  bulan, hari, tahun : integer;  
end;  
var birthdays : array[1..10] of tanggal;
```

Visualisasi Array dari Record



Pemberian nilai elemen birthday

```
Birthdays[1].hari :=25;  
Birthdays[1].bulan:=09;  
Birthdays[1].tahun:=1983;
```

With do...

```
For i:=1 to x do  
  Begin  
    With birthday[i] do  
      Begin  
        Hari:=25;bulan:=09;tahun:=1983;  
      End;  
    End;  
  End;
```

Record di dalam Record

```
type tanggal = record  
  bulan, hari, tahun : integer;  
end;  
type waktu = record  
  jam, menit, detik : integer;  
end;  
type waktu_ini = record  
  tanggal_ini : tanggal;  
  waktu_ini : waktu  
end;  
var saat_ini : waktu_ini;
```

Pemberian Nilai

```
saat_ini.tanggal.bulan:= 11;  
saat_ini.tanggal.hari:= 2;  
saat_ini.tanggal.tahun:= 1985;  
saat_ini.waktu.jam:= 3;  
saat_ini.waktu.menit:= 3;  
saat_ini.waktu.detik:= 33;
```