

ARRAY

Array merupakan kumpulan dari nilai-nilai data yang bertipe sama dalam urutan tertentu yang menggunakan nama yang sama. Dengan menggunakan array, sejumlah variabel dapat memakai nama yang sama. Letak atau posisi dari elemen array ditunjukkan oleh suatu index. Dilihat dari dimensinya array dapat dibagi menjadi Array dimensi satu, array dimensi dua dan array multi-dimensi.

Deklarasi Array

Tipe-Data Nama_Variabel[Ukuran]

dengan :

- **tipe_data** : untuk menyatakan tipe dari elemen array, misalnya int, char, float.
- **nama_var** : nama variabel array
- **ukuran** : untuk menyatakan jumlah maksimal elemen array.

```
int nil[5] = { 1,3,6,12,24 };
```

	0	1	2	3	4
nil	1	3	6	12	24

Mengakses Elemen Array Berdimensi Satu

Pada C, data array akan disimpan dalam memori yang berurutan. Elemen pertama mempunyai indeks bernilai 0. Jika variabel nilai_tes dideklarasikan sebagai array dengan 5 elemen, maka elemen pertama memiliki indeks sama dengan 0, dan elemen terakhir memiliki indeks 4

```
nama_array[index];
```

Pada contoh di atas, variabel nil memiliki 5 buah elemen yang masing-masing berisi data. Pengaksesan tiap-tiap elemen data adalah:

	nil[0]	nil[1]	nil[2]	nil[3]	nil[4]
nil					

Misal, untuk memberikan nilai 75 pada elemen ke 3, maka pernyataannya adalah:

```
nil[2] = 75;
```

atau jika akan memberikan nilai array kepada sebuah variabel a, dapat ditulis:

```
a = nil[2];
```

Array Berdimensi Dua

Array berdimensi satu dapat disimpan pada sebuah array berdimensi dua. Pendeklarasian array berdimensi dua adalah sebagai berikut :

```
int data_lulus[4][3] ;
```

Nilai **4** untuk menyatakan banyaknya **baris** dan **3** menyatakan banyaknya **kolom**

	0	1	2	← indeks kedua (kolom)
0	80	540	1032	
1	15	83	301	
2	8	12	15	
3	10	129	257	

↑ indeks pertama (baris)

int data_lulus[4][3];

Gambar 15.1 Array berdimensi dua

Mengakses Elemen Array Berdimensi Dua

Array seperti data_lulus dapat diakses dalam bentuk

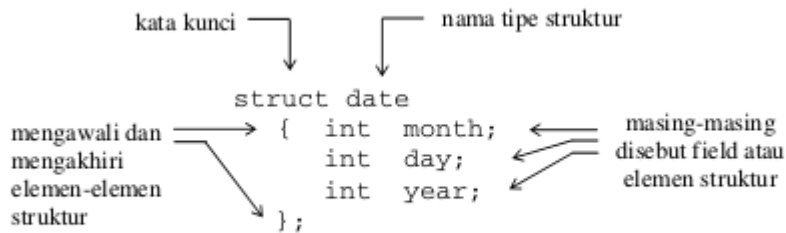
```
data_lulus[indeks pertama, indeks kedua]
```

STRUCT

Struktur adalah pengelompokan variabel-variabel dengan tipe data berbeda dalam satu nama yang sama. Berbeda dengan array yang berisi kumpulan variabel-variabel yang bertipe sama dalam satu nama, maka suatu struktur dapat terdiri atas variabel-variabel yang berbeda tipenya dalam satu nama. Dengan demikian dimungkinkan suatu struktur dapat berisi elemen-elemen data berbeda tipe seperti char, int, float, double, dan lain-lain.

Mendefinisikan & Mendeklarasikan Struct

Keterangan untuk definisi struktur ini dapat dilihat lebih jelas pada gambar 29.1.



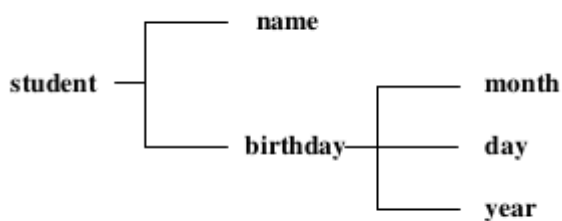
Gambar 29.1 Pendefinisian tipe struktur

```
struct date {
    int month, day, year;
};

struct person {
    char name[30];
    struct date birthday;
};

struct person student;    //deklarasi var student
```

Diagram struktur data dari variabel `student` digambarkan pada gambar 29.3.



Gambar 29.3. Struktur data dari variabel `student`

Mengakses Elemen Struct

Elemen dari suatu variabel struktur dapat diakses dengan menyebutkan nama variabel struktur diikuti dengan operator titik (‘.’) dan nama dari elemen strukturnya.

variabel_struktur.nama_field

TUGAS

1. **Buat Laporan Praktikum**
2. **Buat Program jenjang pendidikan dengan keluaran seperti ini, jenjang pendidikan disesuaikan dengan jenjang pendidikan kalian masing – masing.**

Contoh Output Program :

```
Jenjang pendidikan

Jenjang : SD
Nama Sekolah : SDN Gunung Batu
Alamat Sekolah : Bogor
Tahun Lulus : 2009

Jenjang : SMP
Nama Sekolah : SMP 14 Bogor
Alamat Sekolah : Bogor
Tahun Lulus : 2011

Jenjang : SMA
Nama Sekolah : SMA 9 Bogor
Alamat Sekolah : Bogor
Tahun Lulus : 2014

Process returned 0 (0x0)   execution time : 0.006 s
Press ENTER to continue.
```

Petunjuk :

- Gunakan tipe data struct
- Deklarasikan variabel bertipe array of struct