



01. Review Array, Pointer dan Struktur

ARNA FARIZA
YULIANA SETIOWATI

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

Capaian Pembelajaran

1. Mahasiswa mengerti konsep array, pointer dan struktur.
2. Mahasiswa dapat menjelaskan perbedaan array, pointer dan struktur.
3. Mahasiswa dapat menyelesaikan permasalahan pemrograman menggunakan array pointer dan struktur.



POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

Materi

- Array
- Pointer
- Struktur



Array

- ✓ Array adalah sekumpulan variabel yang mempunyai tipe yang sama.
- ✓ Bersifat statis, dimana alokasi memori dilakukan saat deklarasi source code.
- ✓ Contoh deklarasi array

```
int nilai[10];
```

Tipe data → int
Nama variabel → nilai
Besarnya array → [10]

Terdapat 10 variable nilai yang sama tipenya yaitu integer.



Mengakses Array

- Untuk mengakses array harus ditentukan index yang diakses.

```
void main() {
    int bilangan[10], int i;
    for(i=0;i<10;i++){
        bilangan[i] = 2*i;
        printf("%d %d\n", i, bilangan[i]);
    }
}
```



Pointer

- Pointer adalah variabel yang menunjuk ke lokasi alamat tertentu.
- Bersifat dinamis, dimana alokasi memori dilakukan pada saat program dijalankan.
- Contoh deklarasi pointer

Tipe data
yang ditunjuk
Tanda variable
pointer
Nama variable
pointer

```
int *nilai;
```

Variabel nilai menunjuk ke variable lain yang bertipe integer



Mengakses Pointer

- Terlebih dahulu harus ditentukan alamat yang ditunjuk.

```
void main() {
    int a, int *b;
    a = 5;
    b = &a;
    printf("%d %d\n", a, *b);
}
```

```
void main() {
    char *namaku;
    strcpy(namaku, "Anna");
    printf("%s\n", *nama);
}
```



Array of pointer

- Adalah kumpulan array dengan tipe data pointer
- Contoh deklarasi array of pointer

Tipe data yang ditunjuk → int *nilai[10];

 Tanda variable pointer → *

 Nama variable pointer → nilai

 Jumlah variabel → [10]

Terdapat 10 variable nilai yang sama tipenya yaitu menunjuk ke variable lain yang bertipe integer.



Mengakses array of pointer

```
void main() {
    int *bil[3], a, b, c, i;
    a = 5;
    b = 10;
    c = 15;
    bil[0] = &a;
    bil[1] = &b;
    bil[2] = &c;
    for(i=0;i<3;i++)
        printf("%d\n", *bil[i]);
}
```



Struktur

- Struktur adalah sekumpulan variabel yang mempunyai tipe yang berbeda.
- Sebagai user defined variable, yaitu variable yang didefinisikan sendiri oleh user
- Contoh deklarasi struktur

```
struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
};
```

Struktur tanggal terdiri dari 3 variable dengan tipe yang berbeda.



Deklarasi Struktur dengan typedef

- typedef berfungsi sebagai nama lain / alias dari tipe data.
- Digunakan untuk meringkas tipe data struktur sehingga mudah diakses.

```
typedef struct tanggal TGL;
struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
};
```

```
typedef struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
} TGL;
```



Mengakses Tipe Struktur

- Menggunakan tanda “”

```
typedef struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
} TGL;
void main()
{
    TGL a;
    a.tgl = 17;
    strcpy(a.bulan, "Agustus");
    a.tahun = 1945;
    printf("%d %s %d\n", a.tgl, a.bulan, a.tahun);
}
```



Array of Structure

➤ Tipe data array pada tipe data struktur

```
typedef struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
} TGL;

void main()
{
    TGL tgl_lahir[2];
    int i;
    tgl_lahir[0].tgl = 11;
    strcpy(tgl_lahir[0].bulan, "Maret");
    tgl_lahir[0].tahun = 1965;
    TGL tgl_lahir[2];
    tgl_lahir[1].tgl = 1;
    strcpy(tgl_lahir[1].bulan, "Juni");
    tgl_lahir[1].tahun = 1945;
    for(i=0;i<2;i++)
        printf("%d %s %d\n", tgl_lahir[i].tgl, tgl_lahir[i].bulan, tgl_lahir[i].tahun);
}
```



Pointer to Structure

➤ Variable yang menunjuk ke tipe data struktur

```
typedef struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
} TGL;

void main()
{
    TGL *tgl_lahir;
    tgl_lahir = (TGL *) malloc (sizeof(TGL));
    *(tgl_lahir.tgl) = 11;
    strcpy(*(tgl_lahir.bulan), "Maret");
    *(tgl_lahir.tahun) = 1965;
    printf("%d %s %d\n", *(tgl_lahir.tgl), *(tgl_lahir.bulan), *(tgl_lahir.tahun));
}
```



Pointer to Structure

- Bisa juga dengan cara berikut

```
typedef struct tanggal {
    int tgl;
    char bulan[15];
    int tahun;
} TGL;

void main()
{
    TGL *tgl_lahir;
    tgl_lahir = (TGL *) malloc (sizeof(TGL));
    tgl_lahir->tgl = 11;
    strcpy(tgl_lahir->bulan, "Maret");
    tgl_lahir->tahun = 1965;
    printf("%d %s %d\n", tgl_lahir->tgl, tgl_lahir->bulan, tgl_lahir->tahun);
}
```



Rangkuman

- Sekumpulan variabel yang mempunyai tipe yang sama
- Variabel yang menunjuk ke lokasi alamat tertentu
- Sekumpulan variabel yang mempunyai tipe yang berbeda



Soal

1. Jelaskan perbedaan array, pointer dan struktur.
2. Jelaskan deklarasi array dan cara mengakses array.
3. Jelaskan deklarasi pointer dan cara mengakses pointer.
4. Jelaskan deklarasi struktur dan cara mengakses struktur.

