

Array

Ali Ridho Barakbah

Array

- Array adalah sekelompok data yang bertipe sama yang menggunakan sebuah nama yang sama dan disimpan dalam urutan tertentu.
- Data di suatu array disebut dengan elemen array.
- Letak urutan dari elemen-elemen array ditunjukkan oleh suatu indeks.

Array dimensi 1

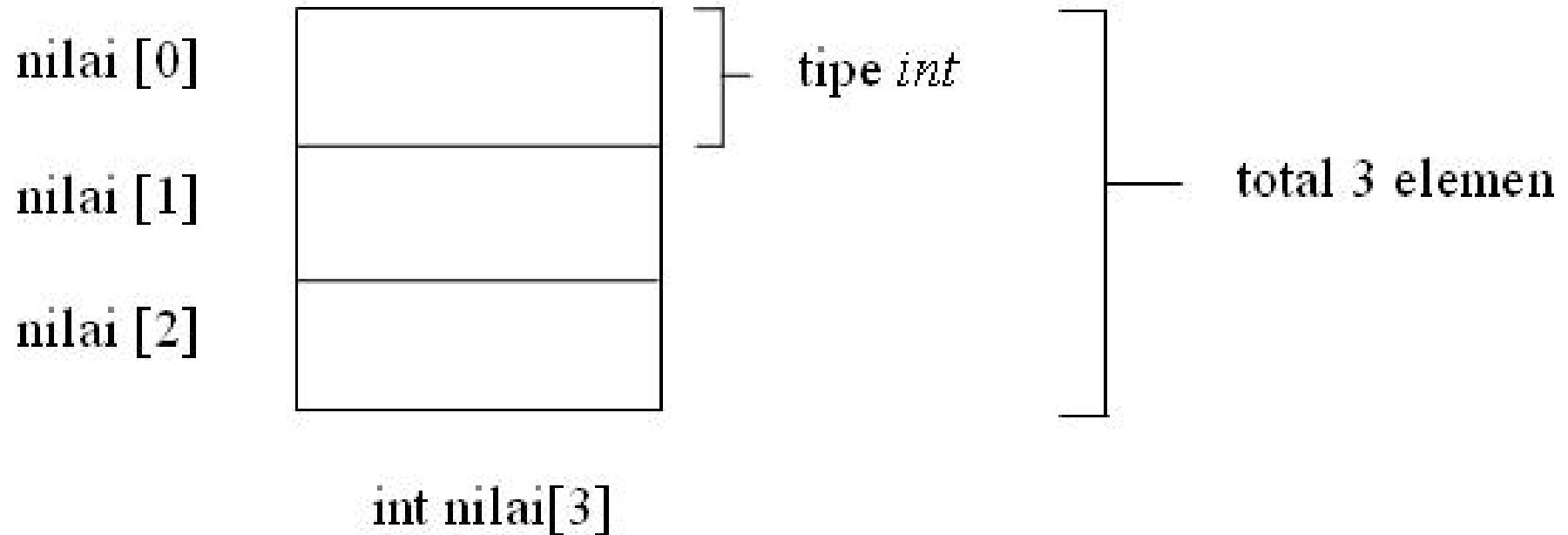
- Deklarasi

```
tipe_data nama_var[panjang_element];
```

- Contoh:

```
int nilai[3];
```

Mengakses elemen array



- Bentuk umum pengaksesan array adalah:

```
nama_var[indeks]
```

nilai[0] → elemen 1, indek ke-0 dari nilai
nilai[2] → elemen 3, indek ke-2 dari nilai

- Contoh:

```
nilai[0] = 70;  
scanf("%d", &nilai[2]);
```

Inisialisasi array

- Inisialisasi array dapat dilakukan dengan cara memasukkan nilai-nilai dari setiap elemen array setelah deklarasi array dan harus diapit dengan tanda { dan }

- Contoh:

```
int jum_hari[12] = { 31, 28, 31, 30, 31, 30,  
                   31, 31, 30, 31, 30, 31};
```

Beberapa variasi dalam deklarasi dan inisialisasi array

- `int numbers[10];`
- `int numbers[10] = { 34, 27, 16 };`
- `int numbers[] = { 2, -3, 45, 79, -14, 5, 9, 28, -1, 0 };`
- `char text[] = "Welcome to New Zealand.";`
- `float radix[12] = { 134.362, 1913.248 };`
- `double radians[1000];`

Array berdimensi 2

- Bentuk deklarasi:

```
tipe_data nama_var[pjg_elm1][pjg_elm2];
```

- Contoh:

```
int nilai[3][5];
```


Mengakses array dimensi 2

- Bentuk umum untuk mengakses elemen array berdimensi 2 adalah:

```
nama_var[indeks_dim1, indeks_dim2]
```

- Contoh:

```
nilai[0][1] = 540;
```

```
printf(“%d”, nilai[1][2]);
```

Inisialisasi array dimensi 2

- Hampir mirip dengan menginisialisasi array berdimensi 1, hanya saja dituliskan pengelompokan nilai dalam tiap dimensi dengan tanda { }.

- Contoh:

```
int nilai[2][3]={ {7, 6, 8},  
                 {5, 7, 6} };
```

Array sebagai parameter

```
int nilai[3];  
...  
proses_data(nilai);
```



```
void proses_data(int x[]) {  
    ...  
}
```