

Keg. Pembelajaran 6 : Larik atau Array [] dalam C++

1. Tujuan Kegiatan Pembelajaran

Setelah mempelajari materi kegiatan pembelajaran ini mahasiswa akan dapat :

- 1) Memahami konsep array dalam pemrogram C++ secara benar.
- 2) Mengenal bentuk array dalam pemrograman baik berdimensi satu, dimensi dua atau dimensi banyak secara benar
- 3) Dapat menggunakan bentuk-bentuk array dalam aplikasinya pada pembuatan program secara tepat
- 4) Dapat mengembangkan bentuk-bentuk array dalam pemrograman secara benar

2. Uraian Materi

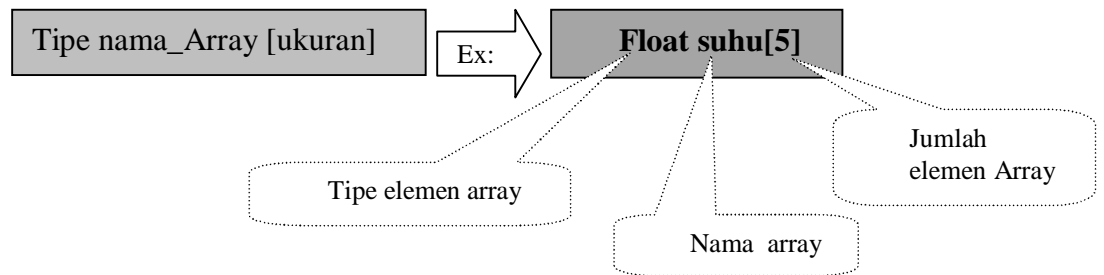
Array merupakan kumpulan dari nilai-nilai data yang bertipe sama dalam urutan tertentu yang menggunakan nama yang sama. Letak atau posisi dari elemen array ditunjukkan oleh suatu index. Dilihat dari dimensinya array dapat dibagi menjadi Array dimensi satu, array dimensi dua dan array multi-dimensi. Beberapa hal yang perlu diperhatikan bahwa array memiliki beberapa sifat sebagai berikut :

- 1) Kumpulan data bertipe sama yang menggunakan nama yang sama
- 2) Sejumlah variabel dapat memakai nama yang sama
- 3) Antara satu variabel dengan variabel lain didalam array dibedakan berdasarkan subscript
- 4) Subscript adalah berupa bilangan di dalam kurung siku []
- 5) Melalui subscript elemen array dapat diakses
- 6) Elemen array tidak lain adalah masing-masing variabel di dalam array

a. Array berdimensi satu

- o Setiap elemen array dapat diakses melalui indeks.
- o Indeks array (subscript) secara default dimulai dari 0.

Bentuk deklarasi array berdimensi satu :



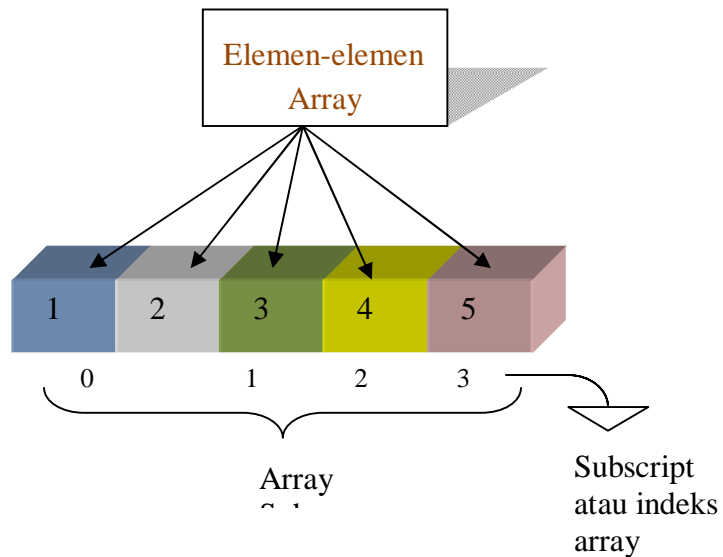
Gambar 26. Bentuk deklarasi array dimensi 1

Tampak bahwa pendefinisian Array meliputi :

- Tipe data elemen array
- Nama array
- Jumlah elemen array

Dengan pendefinisian **float suhu [5];**

Maka array suhu dapat menyimpan lima buah data bertipe float. Hal ini dapat digambarkan sebagai berikut :



Gambar 27. Bentuk Elemen array dimensi 1

Contoh Program :

```
#include <iostream.h>
#include <conio.h>
void main ()
{
    int i;
    float suhu [5]; //array dengan 5 elemen bertipe float
    clrscr ();
    //membaca dari keyboard dan meletakkan ke array
    cout <<"masukkan 5 buah data suhu" <<endl;
    for (i = 0; i<5; i++)
    {
        cout<< i+1<<" : ";
        cin>> suhu [i];
    }
    //menampilkan isi array ke layar
    cout <<"data suhu yang anda masukkan : " << endl;
    for (i=0; i<5; i++)
    cout <<suhu[i]<<endl;
}
```

✚ Mengakses Elemen Array

Setelah suatu array didefinisikan, elemen array dapat diakses dengan bentuk :

Nama_array [subscript]

Dari gambar array di atas, kotak nomor 3 dari kiri dapat diakses dengan menyebutkan :

suhu [2]

Atau dalam bentuk yang lebih umum

suhu [i]

Menyatakan "elemen suhu dengan indeks sama dengan **i**".

Perintah seperti :

cin>> suhu[i]

Berarti "membaca data dari keyboard dan meletakkan di elemen nomor **i** pada array suhu".

Adapun pernyataan seperti

```
cout << suhu [i];
```

Akan menampilkan elemen bernomor **i** pada array suhu.

Contoh program yang melakukan pemrosesan terhadap elemen array sbb:

```
//Program pemrosesan array: Menghitung rerata suhu
#include <iostream.h>
#include <conio.h>
const int JUM_DATA = 5;
void main ()
{
    float suhu[JUM_DATA]; //Array suhu
    float total;
    int i;
    clrscr ();
    //membaca dari keyboard dan meletakkan ke array
    cout <<"masukkan data suhu" <<endl;
    for (i = 0; i<JUM_DATA; i++)
    {
        cout<< i+1<<" : ";
        cin>> suhu [i];
    }
    //menghitung Nilai Rerata
    total = 0;
    for (i=0; i<JUM_DATA; i++)
        total +=suhu[i]; //tambahkan isi suhu [i] ke total
    cout <<" Suhu Rerata = "<<total/JUM_DATA<<endl;
}
```

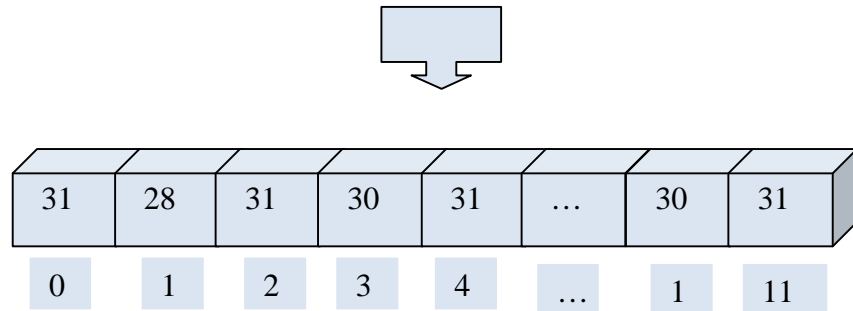
Catatan :

Jangan sampai melakukan pengaksesan elemen di luar jumlah elemen yang didefinisikan, misalnya `float suhu [4];` Harus dipastikan bahwa tidak ada pengaksesan elemen suhu bernomor di luar 0 sampai dengan 3. Hal ini dapat menyebabkan data atau array lain berubah. C++ tidak akan memberikan pesan kesalahan kalau anda melakukan pengaksesan dengan nilai indeks yang berada di luar nilai yang telah didefinisikan.

✚ Memberikan nilai awal thd Array (Inisialisasi)

Misalnya :

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



Gambar 28. Bentuk array dimensi 1 untuk pemberian nilai awal

Berikut Contoh Program yang memberikan inisialisasi terhadap Array berdimensi satu:

```
#include <iostream.h>
#include <conio.h>
#include <iomanip.h>

void main ()
{
    int i;
        //pendefinisian array jum_hari dan
        // pemberian nilai awal

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

    clrscr ();

        //tampilkan isi jum_hari
    for (i = 0; i<12; i++)
        cout << "Jumlah Hari [" <<i<<" ] = " << jum_hari [i] <<
endl;
}
```

Apabila Suatu array diinisialisasi saat didefinisikan, jumlah elemen array boleh tidak disebutkan. Scr otomatis kompiler akan menghitung sendiri, misalnya :

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

Dapat disederhanakan menjadi :

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

Untuk memberikan nilai 0 thd seluruh elemen array pada saat didefinisikan, kita dapat memberikan nilai 0 thd elemen pertama.

Contoh :

```
int tmp[100] = {0};
```

Artinya sama saja dengan memberikan nilai nol dari indeks bernilai 0 hingga 99.

b. ARRAY DIMENSI DUA

Array dua dimensi merupakan array yang terdiri dari m buah baris dan n buah kolom. Bentuknya dapat berupa matriks atau tabel. Digunakan untuk berbagai keperluan, misalnya dalam bentuk tabel, matrik dan sebagainya

Bentuk Umum Array Dua Dimensi :

```
tipe_larik nama_larik [ukuran1][ukuran2]
```

Perhatikan contoh berikut ini :

Tabel 11. Data Kelulusan Jurusan Diknik Elektro

No	Program Studi	1992	1993	1994	1995
1	Diknik Elektro S1	35	45	80	120
2	Diknik Mekatronik	110	110	70	101
3	Teknik elektro D3	10	15	20	17

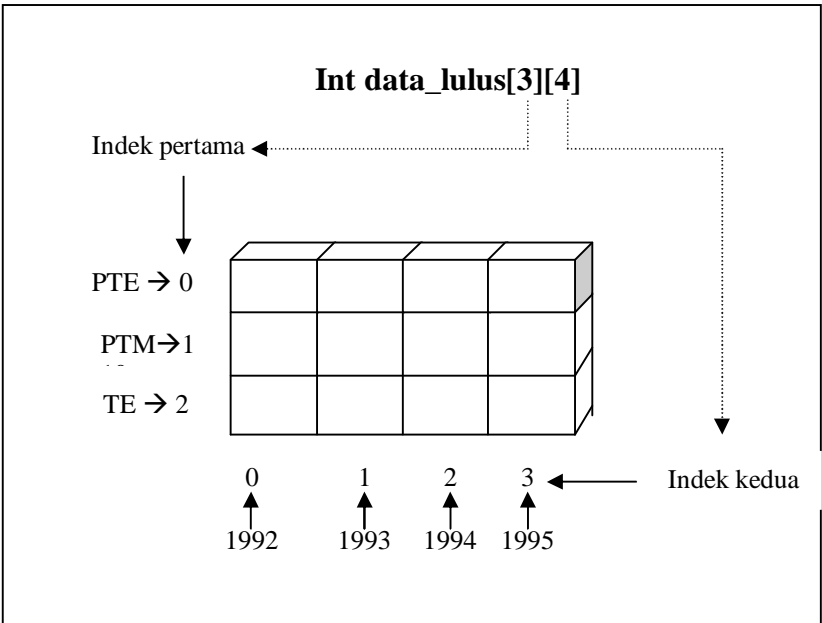
Mendefinisikan Array berdimensi dua
Dari tabel diatas dapat dituangkan ke dalam array dimensi dua.
Pendefinisianya sbb:

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

Pada definisian di atas:

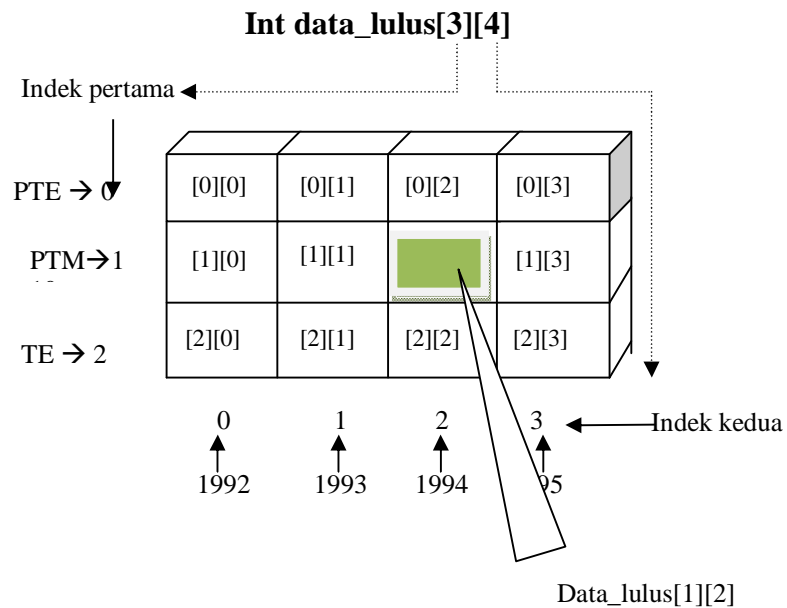
- ✚ Angka 3 menyatakan jumlah baris (mewakili program studi)
- ✚ Angka 4 menyatakan jumlah kolom (mewakili tahun kelulusan)

Dapat digambarkan seperti ini :



Gambar 29. Bentuk umum array dimensi 2

Samahalnya dengan array berdimensi satu, data pada array dimensi dua ditempatkan di memeori yang berurutan. Array berdimensi dua , perhatikan gambar :



Gambar 30. Isi data pada array dimensi 2

Cara mengakses elemen di dalam array berdimensi dua digunakan dalam bentuk :

```

Nama_array [indeks_baris,
indeks_kolom]

```

Contoh :

```
1. data_lulus[1][2] = 70;
```

Merupakan instruksi untuk memberikan nilai 70 ke baris 1 kolom 2 (baris dan kolom diawali dari 0)

```
2. cout << data_lulus[1][2];
```

Merupakan perintah untuk menampilkan elemen data_lulus dengan indeks pertama (baris) berupa 1 dan ineks kedua (kolom) bernilai 2.

Contoh program : Pemakaian Array dimensi dua untuk menampung data kelulusan dari tiga buah prodi untuk 4 tahun terakhir:

```
#include <iostream.h>
#include <conio.h>
void main ()
{
```



```

int data_lulus[3][4]
int tahun, prodi;
clrscr();
        //memberikan data ke elemen array data_lulusan
data_lulus[0][0]      = 35;      //data lulus PTE = 1992
data_lulus[0][1]      = 45;      //data lulus PTE = 1993
data_lulus[0][2]      = 90;      //data lulus PTE = 1994
data_lulus[0][3]      = 120;     //data lulus PTM =1995
data_lulus[1][0]      = 100;     //data lulus PTM= 1992
data_lulus[1][1]      = 110;     //data lulus PTM= 1993
data_lulus[1][2]      = 70;      //data lulus PTM = 1994
data_lulus[1][3]      = 101;     //data lulus PTE = 1995
data_lulus[2][0]      = 10;      //data lulus TE = 1992
data_lulus[2][1]      = 15;      //data lulus TE = 1993
data_lulus[2][2]      = 20;      //data lulus TE = 1994
data_lulus[2][3]      = 17;      //data lulus TE = 1995
        //proses untuk memperoleh informasi lulusan
while (1)
{
    cout <<" Prodi ( 0 = PTE  1=PTM   2=TE :";
    cin >> prodi;
    if ((prodi ==0) || (prodi ==1) || (prodi ==2))
        break;          //keluar dari while
}
while (1)
{
    cout << "Tahun (1992 – 1995): ";
    cin >> tahun;
    if ( (tahun >= 1992) && (tahun <= 1995) )
    {
        tahun -= 1992;      //konversi ke 0, 1, 2,atau 3
        break;              //keluar dari while
    }
}
cout <<"Jumlah yang lulus = " <<data_lulus[prodi][tahun] <<
endl;
}

```

Perhatikan :

- Tanda kurung [] digunakan untuk menunjukkan elemen larik
- Perhitungan elemen larik dimulai dari 0, bukan 1

C++ tidak mengecek larik. Bila anda menyatakan `int x[10]`, ini artinya 10 elemen yang dimulai dari 0. Karena itu elemen terakhir larik adalah `x[9]`. Bila anda salah mereferensikannya dengan `x[10]`, anda akan mendapatkan harga yang tidak terpakai. Akan lebih buruk lagi jika anda memberikan harga ke `x[10]`, yang tidak dapat diterima.

Array dimensi dua mewakili bentuk suatu matrik atau tabel. Untuk matrik misalnya, suatu matrik X sbb :

$$X = \begin{matrix} 8 & 5 & 9 & 7 \\ 8 & 2 & 1 & 0 \end{matrix}$$

Bentuk matrik ini dapat ditulis :

```
int X[2][4];
```

Jika ingin dideklarasikan dengan memberikan nilai-nilai elemennya dapat dituliskan sbb:

```
int X[2][4] = { 8, 5, 9, 7, 4, 2, 1, 0 };
```

Atau untuk kejelasan dapat dituliskan :

```
int X[2][4] = {
                8, 5, 9, 7,
                4, 2, 1, 0
            }
```

Atau untuk dapat dengan bentuk array dengan tak berukuran sbb:

```
int X[][4] = { 8, 5, 9, 7, 4, 2, 1, 0};
```

Dalam bentuk matrik dua dimensi ini, dimensi pertama menunjukkan baris, dan dimensi kedua menunjukkan kolomnya. Elemen dari dua dimensi X dapat dituliskan :

$$\begin{matrix} X[0][0] = 8 & X[0][1] = 5 & X[0][2] = 9 & X[0][3] = 7 \\ X[1][0] = 4 & X[1][1] = 2 & X[1][2] = 1 & X[1][3] = 0 \end{matrix}$$

Program matrik dapat dituliskan sebagai berikut :

```
//Program matrik
#include <iostream.h>
#include <conio.h>
void main ()
{
    int I,J;
```

```

float X[3][4]= {
                12.34, 25.36, 17.45, 18.37,
                15.15, 18.08, 11.25, 55.55,
                43.21, 17.11, 62.85, 13.01
                };

//Masukkan data matrik A
cout <<"masukkan matrik A: "<<endl;
for (l=0; l<3; l++)
{
    for (J=0; J<4; J++)
        cout << setw(6) <<X[l][J];
        cout<<' ' << endl;
    }
}

```

c. Array Multi-Dimensi

Array multi-dimensi merupakan array yang mempunyai ukuran lebih dari dua. Bentuk pendeklarasian array sama saja dengan array dimensi satu maupun array dimensi dua.

Bentuk umum pendefinisian array berdimensi **tiga** :

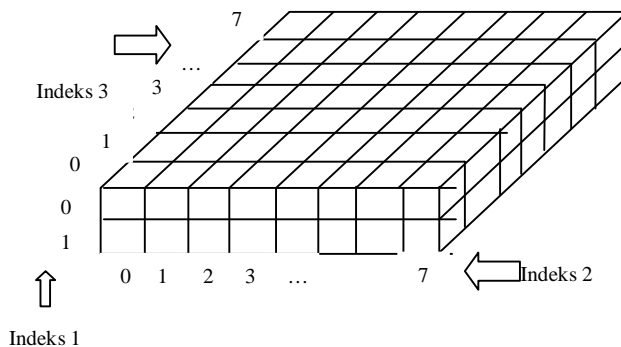
Tipe nama_array[indeks1][indeks2][indeks3];

Contoh :

```
int huruf[2][8][8];
```

Merupakan pendfinisia data huruf sebagai array berdimensi tiga. Gambaran array berdimensi tiga sebagai berikut :

Int Huruf[2][8][8];



Gambar 31. Array Dimensi tiga

3. Rangkuman

Array memiliki beberapa cirri-ciri khusus antara lain bahwa array merupakan kumpulan data bertipe sama yang menggunakan nama yang sama. Sejumlah variabel dapat memakai nama yang sama. Antara satu variabel dengan variabel lain didalam array dibedakan berdasarkan subscript. Subscript adalah berupa bilangan di dalam kurung siku []. Melalui subscript elemen array dapat diakses. Elemen array tidak lain adalah masing-masing variabel di dalam array.

Array dapat dibedakan menjadi beberapa ukuran atau dimensi, yaitu array berdimensi satu, dimensi dua dan multi dimensi (dimensi banyak). Pada array berdimensi satu digunakan pada kumpulan data bertipe sama yang menggunakan nama yang sama, misalnya array bernama bunga = { melati, mawar, dahlia..}. Array dimensi dua diimplementasikan pada pembuatan matrik. Array dimensi banyak dalam aplikasinya tidak terlalu sering dipakai. Namun demikian tidak berarti array dimensi banyak tidak dipelajari.

4. Latihan

1). Buatlah algoritma dan program untuk menyimpan data berikut ke dalam larik :

10 4 2 5 3 8 9 2 9 5

kemudian carilah bilangan yang terbesar !

2). Buatlah algoritma dan program untuk membaca data secara berulang dari keyboard dan meletakkannya ke dalam suatu larik. Jumlah maksimal yang dapat dimasukkan ke dalam larik adalah 10 buah. Setelah itu tampilkan seluruh data yang dimasukkan dari keyboard tadi.

3). Buatlah program untuk melakukan penjumlahan dan pengurangan dua buah matrik A dan Matrik B.