**Tujuan Praktikum:**

Setelah melaksanakan praktikum ini mahasiswa diharapkan dapat menggunakan array statik berdimensi-n dalam berbagai kasus pemrograman.

**Dasar Teori:**

## Deklarasi Array

Array digunakan untuk menyimpan banyak elemen yang sama jenis atau tipenya. Setiap elemen pada array tersebut di akses dengan sebuah indeks. *Array* didefinisikan dengan sebuah nama, tipe data, dan jumlah elemen di dalam *array* tersebut dengan bentuk:

**tipe\_data** namaArray [**jumlah\_elemen**];

Masing-masing elemen dapat diakses dengan menambahkan indeks pada variabel tersebut yaitu: **nama\_array[indeks]**. Berbeda dengan bahasa lainnya, indeks awal *array* pada Bahasa C++ dimulai dari 0 (nol). Selain menggunakan konstanta, indeks *array* dapat juga berupa suatu variabel ordinal (integer, char, atau enumerasi).

*Array* dapt dibedakan atas jumlah dimensinya, *array* berdimensi satu disebut ***vektor***, array berdimensi dua disebut ***matriks*** dan *array* berdimensi lebih dari dua disebut ***tensor*.** *Array* dua dimensidideklarasikan dengan bentuk:

**tipe\_data** NamaArray [**jumlah\_baris**][**jumlah\_kolom**];

Deklarasi ini menyatakan bahwa NamaArray adalah *array* satu-dimensi dengan jumlah elemen jumlah\_baris, dimana masing-masing elemennya adalah sebuah *array* satu-dimensi dengan jumlah elemen jumlah\_kolom.

**Inisialisasi**

*Array* diinisialisasi dengan menuliskan deretan nilai diantara simbol { dan } seperti berikut:

**float** data[**5**] = { 0.1, 0.4, 0.6, 0.8, 0.9 };

*/\* data adalah array bertipe float dengan jumlah elemen 5*

*yaitu 0.1, 0.4, 0.6, 0.8, dan 0.9 \*/*

**char** vokal[**5**] = { 'a', 'i', 'u', 'e', 'o' };

*/\* vokal adalah array bertipe karakter dengan jumlah*

*elemen 5 yaitu 'a', 'i', 'u', 'e', dan 'o' \*/*

**Operasi Pada *Array***

Dua operasi terpenting terhadap suatu *array* adalah Pencarian (*searching*) dan Pengurutan (*sorting*). Banyak sekali metode yang dapat digunakan untuk kedua operasi ini, salah satu teknik pencarian yang paling sederhana adalah *linear search*, yaitu mencari suatu data di dalam suatu *array* dengan cara menelusuri elemen *array* satu per satu dari awal sampai akhir.

***Array* sebagai Parameter Fungsi**

*Array* dapat digunakan sebagai parameter fungsi dengan menggunakan bantuan pointer. Dua jenis sintaks yang sering digunakan adalah sebagai berikut :

**void** fungsi1 (**int** array[]);

Jika besar *array* sudah dapat dipastikan, boleh juga digunakan bentuk seperti:

**void** fungsi1 (**int** array[100]);

Dalam fungsi ini elemen *array* dapat diakses seperti biasanya. Sedangkan untuk mengirimkan *array* kedalam fungsi tersebut, cukup dituliskan nama *array* yang akan dikirim ke dalam parameter fungsi tersebut.

*Array* berdimensi banyak dapat dikirimkan kedalam fungsi dengan cara yang mirip dengan *array* satu dimensi, tetapi besar semua dimensi *array*, kecuali dimensi pertama, harus dideklarasikan pada parameter fungsi, seperti pada contoh berikut:

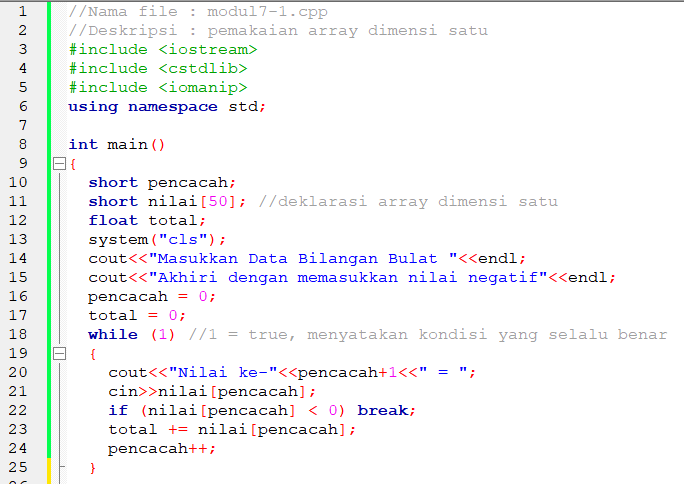
**void** fungsi2 ( **float** matrix[][5] ) ;

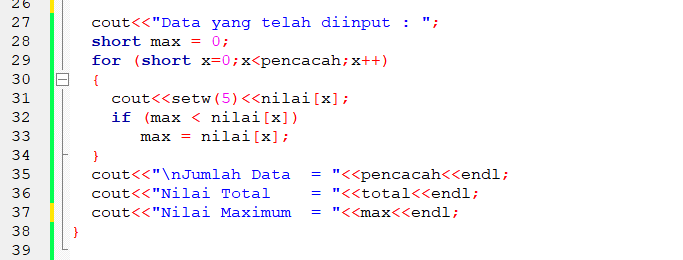
atau

**void** fungsi2 ( **float** matrix[6][5] ) ;

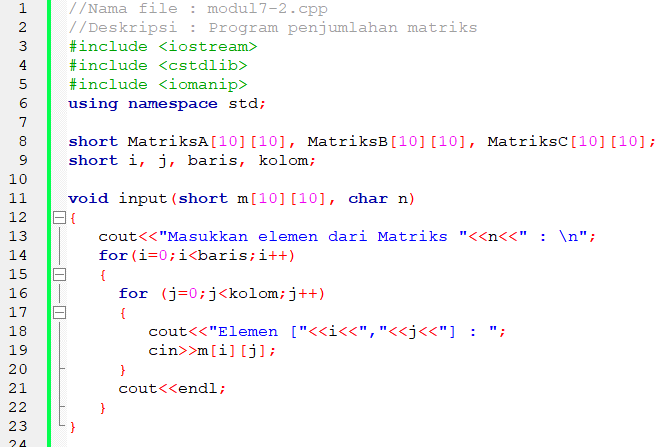
**Kegiatan Praktikum:**

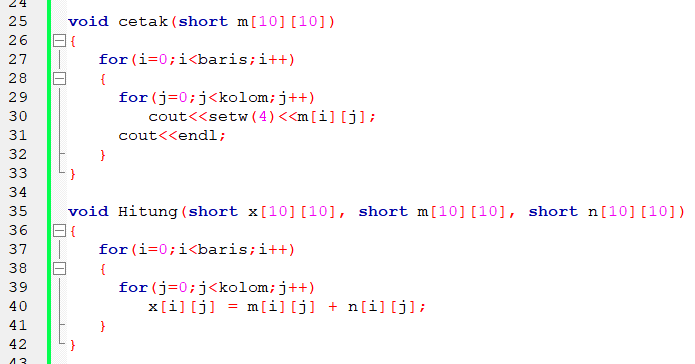
1. Program berikut menunjukkan penggunaan array berdimensi satu untuk menerima sejumlah data bilangan bulat, menghitung total jumlahannya serta menampilkan kembali seluruh data bilangan tesebut dan menampilkan nilai maksimumnya.

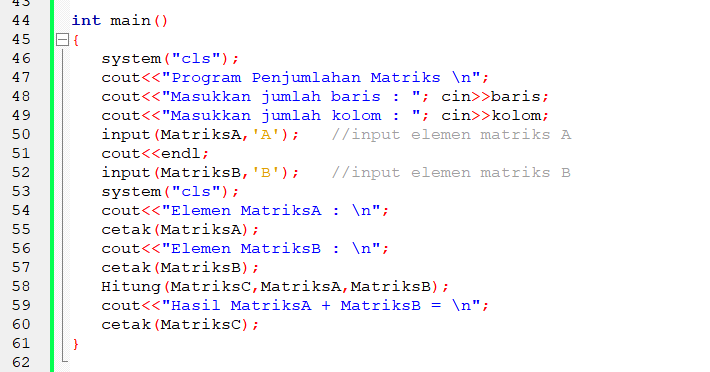




1. Program berikut menunjukkan penggunaan array berdimensi dua untuk melaku-kan proses penjumlahan terhadap dua buah matriks.



****

****