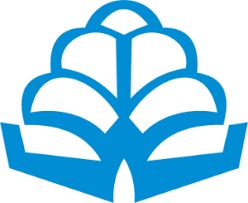
**POKJAR 2 (ALGORITMA PEMROGRAMAN)**

**ARRAY**

****

**Disusun Oleh :**

**Kelompok 5 (Lima)**

1. **Fadhilah Puspitasari (24031009**
2. **Diah Merlinda Rifandy (24031007)**
3. **Maria Hawa Lulua Salsabila (24031018)**
4. **Yasmin Salsabila (24031050)**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI TEKNOLOGI INDONESIA**

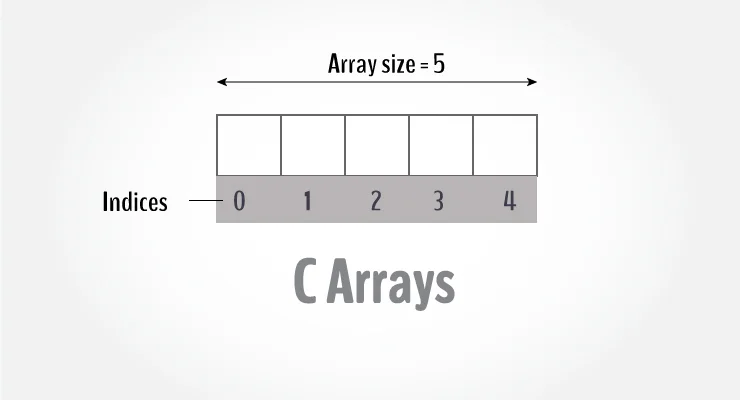
ARRAY

1. Pengertian

Array adalah kumpulan-kumpulan variabel yang menyimpan data dengan tipe yang sama atau data-data yang tersusun secara linear dimana di dalamnya terdapat elemen dengan tipe yang sama. Indeks dalam array menyatakan elemen yang disimpan dan panjang atau length menyatakan total elemen yang tersimpan.

Indeks dari elemen array baik bahasa Java maupun C++ dimulai dari 0, bukan 1. Dalam array, untuk membedakan satu variabel dengan variabel lain berdasarkan subscript, bilangan dalam kurung siku […] disebut subscript, dengan subscript masing-masing elemen dapat diakses.

Dalam menyusun fungsi array ada tiga struktur, yaitu kumpulan data dengan tipe sama, gunakan indeks untuk mengakses setiap elemen, dan simpan di tempat yang bersambungan.



1. Mendeklarasikan Array

Suatu array berdimensi satu dideklarasikan dalam bentuk umum berupa :

tipe\_data nama\_var[ukuran];

dengan :

tipe\_data : untuk menyatakan tipe dari elemen array, misalnya int, char, float.

nama\_var : nama variabel array

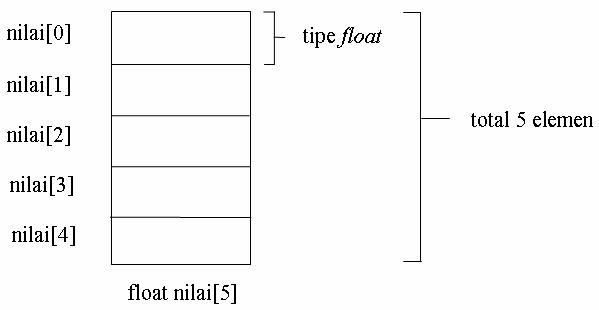
ukuran : untuk menyatakan jumlah maksimal elemen array.

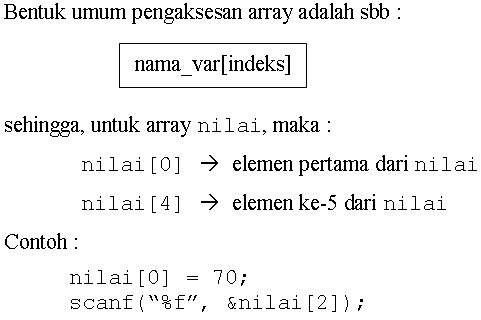
Contoh pendeklarasian array :

float nilai[5];

menyatakan bahwa variabel nilai bertipe array of float dan memiliki 5 elemen bertipe float.

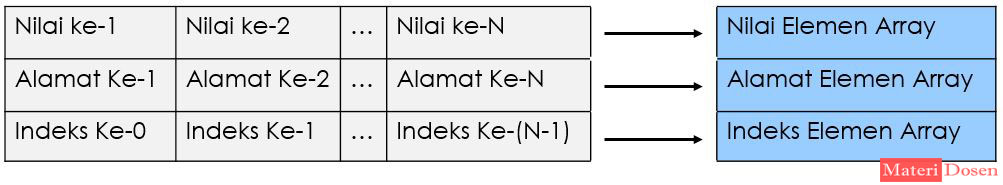
1. Mengakses Elemen Array
2. Pada C++, data array akan disimpan dalam memori yang berurutan.
3. Elemen pertama mempunyai indeks bernilai 0.
4. Jika nilai dideklarasikan sebagai array of float dengan 5 elemen, maka elemen pertama memiliki indeks sama dengan 0, dan elemen terakhir memiliki indeks 4.

`



1. Array Dimensi 1
2. Pengertian

Array satu dimensi adalah array yang terdiri dari n buah kolom atau Array satu dimensi adalah array yang terdiri dari 1 subskrip array saja. Setiap elemen array satu dimensi dapat diakses melalui indeks yang terdapat di dalamnya.

Berikut gambar yang dapat mengilustrasikan sebuah array satu dimensi.

1. Bentuk / Cara Pendeklarasian Array Satu Dimensi

Untuk mendeklarasikan sebuah array satu dimensi dalam C++, sobat harus menggunakan tanda [ ] (bracket). Adapun bentuk umum dari pendeklarasian array satu dimensi adalah sebagai berikut:

tipe\_data nama\_array [jumlah\_elemen];

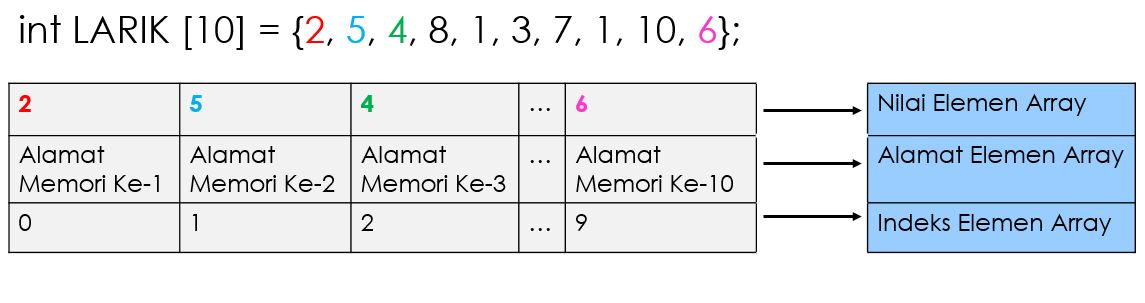
Sebagai contoh misal kita ingin mendeklarasikan sebuah array satu dimensi (misalnya dengan nama LARIK) yang mempunyai 10 elemen dengan tipe data int, maka bentuk array nya adalah sebagai berikut:

int LARIK [10];

Seandainya array LARIK akan kita beri nilai saat pendeklarasian (inisialisasi), maka contoh penulisannya adalah sebagai berikut:

int LARIK [10] = {2, 5, 4, 8, 1, 3, 7, 1, 10, 6};

Dari pendeklarasian sekaligus inisialisasi array LARIK diatas, dapat di artikan bahwa kita telah memesan tempat pada memori komputer sebanyak 10 tempat, dengan indeks dari 0 sampai 9, dimana nilai - nilai akan dimasukan ke elemen array secara berturut - turut, mulai dari indeks 0 akan di isi dengan nilai '2' sampai indeks 9 yang di isi dengan nilai '6'. Untuk lebih jelasnya berikut kami sajikan ilustrasi dari array LARIK.

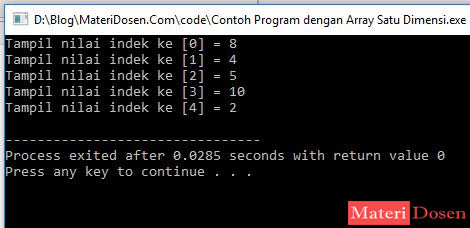


Perlu diketahui, ruang memori yang dibutuhkan untuk deklarasi array tersebut adalah 40 byte, yang berasal dari 10 x 4 byte (4 merupakan ukuran tipe data int sedangkan 10 merupakan jumlah elemen array).

1. Contoh Program Dengan Array Satu Dimensi Dalam C++

Berikut ini adalah contoh program menggunakan Array Satu Dimensi yang didalamnya terdapat proses deklarasi sekaligus inisialisasi array (pengisian elemen array) dan proses menampilkan nilai yang terdapat pada elemen array.

1. Contoh pertama



#include <iostream>

using namespace std;

int main(){

// Deklarasi sekaligus inisialisasi (pengisian awal)

// Array 'ARnilai' dengan 5 buah elemen berisi int

int ARnilai[5]={8,4,5,10,2};

// Karena sudah melakukan inisialisasi maka kita

// Tidak wajib mengisi nilai kedalam Elemen Array

// Menampilkan nilai atau isi dari array 'ARnilai'

for(int i=0; i<5;i++){

cout<<"Tampil nilai indek ke ["<<i<<"] = "<<ARnilai[i];

cout<<endl;

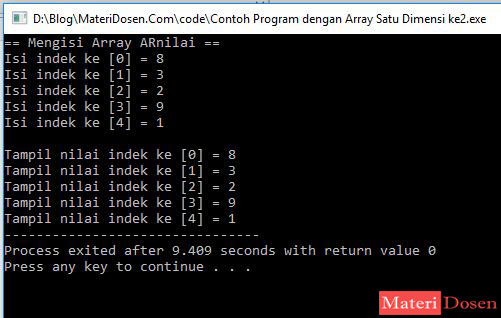
}

return 0;

}

1. Contoh kedua

Dibawah ini adalah contoh program menggunakan Array Satu Dimensi yang didalamnya terdapat proses mengisi nilai ke dalam elemen array dan proses menampilkan nilai yang terdapat pada elemen array.



#include <iostream>

using namespace std;

int main(){

// Deklarasi array 'ARnilai' dengan 5 buah elemen berisi int

int ARnilai[5];

// Mengisi nilai ke dalam elemen array

cout<<"== Mengisi Array ARnilai ==\n";

for(int i=0; i<5;i++){

cout<<"Isi indek ke ["<<i<<"] = ";

cin>>ARnilai[i];

}

// Menampilkan nilai atau isi dari array 'ARnilai'

for(int i=0; i<5;i++){

cout<<"\nTampil nilai indek ke ["<<i<<"] = "<<ARnilai[i];

}

return 0;

}

1. Array Dimensi 2
2. Pengertian

Array dua dimensi adalah sebutan untuk array yang penomoran index-nya menggunakan 2 buah angka. Analogi yang sering dipakai seperti titik koordinat dalam diagram kartesius.

Diagram kartesius merupakan diagram yang biasa kita pakai untuk membuat grafik. Disini terdapat sumbu X dan sumbu Y. Sebuah titik dalam diagram kartesius ini harus disebut secara berpasangan, seperti (2,3) atau (-3, 1).

Analogi lain adalah matriks. Dalam matematika, matriks terdiri dari kolom dan baris. Kembali, untuk menentukan nilai dari sebuah matriks, kita harus sebut secara berpasangan seperti baris 1 kolom 1, atau baris 2 kolom 3, dst. Konsep seperti inilah yang menjadi dasar dari array 2 dimensi.

Untuk membuat array 2 dimensi di dalam bahasa C++, caranya tulis 2 kali tanda kurung siku setelah nama variabel, seperti contoh berikut:

int arr[2][2];

Baris diatas akan membuat array 2 dimensi dengan nama variabel: arr. Variabel arr ini total berisi 4 element (2 x 2). Atau jika diibaratkan sebagai matriks, disini kita membuat matriks 2 x 2.

Untuk mengakses setiap element array, penulisan index juga harus ditulis 2 kali, seperti contoh berikut:

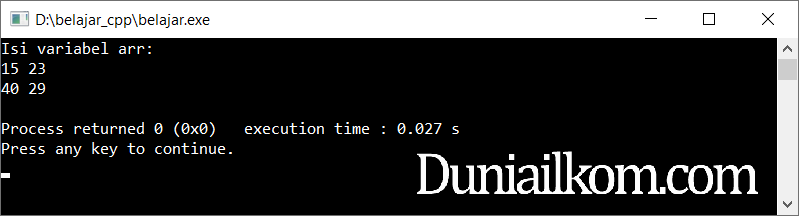
arr[0][0] = 20;

arr[0][1] = 58;

arr[1][0] = 22;

arr[1][1] = 98;

1. Berikut contoh kode program pertama dari pembuatan array 2 dimensi di dalam bahasa C++:



#include <iostream>

using namespace std;

int main()

{

int arr[2][2];

arr[0][0] = 15;

arr[0][1] = 23;

arr[1][0] = 40;

arr[1][1] = 29;

cout << "Isi variabel arr:" << endl;

cout << arr[0][0] <<" "<< arr[0][1] << endl;

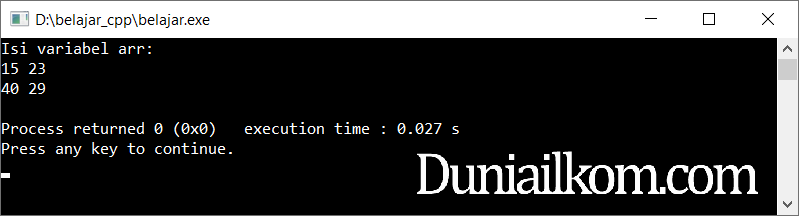
cout << arr[1][0] <<" "<< arr[1][1] << endl;

return 0;

}

1. Contoh 2

Di awal kode program, array arr dengan perintah int arr[2][2];. Artinya, array arr adalah sebuah array 2 dimensi dengan 4 element. Proses pengisian setiap element array dilakukan setelahnya, kemudian ditampilkan dengan perintah cout.Pengisian element array 2 dimensi juga bisa dilakukan pada saat array itu di definisikan, seperti contoh berikut:



#include <iostream>

using namespace std;

int main()

{

int arr[2][3] = {{10,20,30},{33,44,55}};

cout << "Isi variabel arr:" << endl;

cout << arr[0][0] <<" "<< arr[0][1] <<" "<< arr[0][2]<< endl;

cout << arr[1][0] <<" "<< arr[1][1] <<" "<< arr[1][2]<< endl;

return 0;

}

1. Contoh study kasus

Buat matriks 2 dimensi, inputkan datanya kemudian kalikan dengan bilangan 5.

#include <iostream>

using namespace std;

int main() {

int A[2][2];

int C[2][2];

int baris = 2, kolom = 2;

// Input elemen matriks A

cout << "Masukkan elemen matriks A:" << endl;

for (int i = 0; i < baris; i++) {

for (int j = 0; j < kolom; j++) {

cin >> A[i][j];

}

}

// Penjumlahan matriks

for (int i = 0; i < baris; i++) {

for (int j = 0; j < kolom; j++) {

C[i][j] = A[i][j]\* 5;

}

}

// Tampilkan matriks A

cout << "Matriks A:" << endl;

for (int i = 0; i < baris; i++) {

for (int j = 0; j < kolom; j++) {

cout << A[i][j] << " ";

}

cout << endl;

}

// Tampilkan matriks C hasil perkalian 5

cout << "Matriks C (hasil perkalian 5):" << endl;

for (int i = 0; i < baris; i++) {

for (int j = 0; j < kolom; j++) {

cout << C[i][j] << " ";

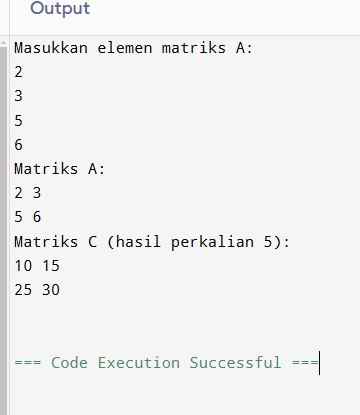
}

cout << endl;

}

return 0;

}



1. Latihan Soal
2. Deklarasikan sebuah variabel array (misalkan nama variabelnya = **letters**) yang mengalokasikan untuk 10 elemen bertipe *char*. Kemudian masukkan karakter ‘Z’ pada elemen yang ke-empat dari array **letters.**

|  |
| --- |
| #include <iostream>  using namespace std;  int main(){  int nilai [10];  int i;  char letters [10];  letters [3] = 'Z';  cout << "Karakter Elemen Ke-4 adalah : " << letters [3] << endl;      return 0;  } |

|  |
| --- |
| Karakter Elemen Ke-4 adalah : Z  === Code Execution Successful === |

1. Deklarasikan sebuah array bertipe char dengan panjang 10. Kemudian cetak ke layar isi array tersebut dengan indeks ganjil.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  char arr[10]= {'a','b','c','d','e','f','g','h','i','j'};  int i;    cout<<"isi array dengan indeks ganjil: "<<endl;  for (int i=1; i<10; i+=2){  cout<<"indeks "<<i<<";"<<arr[i]<<endl;  }      return 0;  } |

|  |
| --- |
| isi array dengan indeks ganjil:  indeks 1;b  indeks 3;d  indeks 5;f  indeks 7;h  indeks 9;j  === Code Execution Successful === |

1. Deklarasikan sebuah array bertipe char dengan panjang 20. Kemudian inputkan sebuah karakter dari keyboard dan cari apakah karakter tersebut ada dalam array yang telah dideklarasikan.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  char arr[20];  char karakter;  bool ditemukan = false;  cout << "Masukkan karakter yang ingin dicari: ";  cin >> karakter;    cout << "Masukkan 20 karakter untuk mengisi array:" <<endl;  for (int i = 0; i < 20; i++) {  cout << "Karakter ke-" << i + 1 << ": ";  cin >> arr[i];  }  for (int i = 0; i < 20; i++) {  if (arr[i] == karakter) {  ditemukan = true;  cout << "Karakter '" << karakter<< "' ditemukan pada indeks ke-" << i <<endl;  }  }    if (!ditemukan) {  cout << "Karakter '" << karakter<< "' tidak ditemukan dalam array." <<endl;  }  return 0;  } |

|  |
| --- |
| Masukkan karakter yang ingin dicari: p  Masukkan 20 karakter untuk mengisi array:  Karakter ke-1: a  Karakter ke-2: a  Karakter ke-3: p  Karakter ke-4: p  Karakter ke-5: s  Karakter ke-6: s  Karakter ke-7: s  Karakter ke-8: s  Karakter ke-9: s  Karakter ke-10: p  Karakter ke-11: g  Karakter ke-12: g  Karakter ke-13: g  Karakter ke-14: p  Karakter ke-15: g  Karakter ke-16: g  Karakter ke-17: a  Karakter ke-18: a  Karakter ke-19: p  Karakter ke-20: a  Karakter 'p' ditemukan pada indeks ke-2  Karakter 'p' ditemukan pada indeks ke-3  Karakter 'p' ditemukan pada indeks ke-9  Karakter 'p' ditemukan pada indeks ke-13  Karakter 'p' ditemukan pada indeks ke-18  === Code Execution Successful === |

1. Buat program untuk mencari nilai rata rata seorang mahasiswa

Input :

Nama : Agus

Jumlah nilai : 3

Nilai 1 : 60

Nilai 2 : 70

Nilai 3 : 80

Output : Agus, nilai rata rata 70

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {     int i, nilai [2], angka, sum, average;     int ratarata, total = 0;     cout << "Menghitung Nilai Rata-rata Mahasiswa" << endl << endl;     cout << "Nama : Agus" << endl;     cout << "Jumlah Nilai : 3" << endl;     //pengisian elemen array     for (i = 0; i < 3; i++) {      cout << "Nilai Ke-" << i+1 << " : ";      cin >> nilai [i];      sum += nilai [i];     }      ratarata = sum/3;      //menampilkan data ke layar      cout << endl;      cout << "Nilai Rata-rata adalah : " << ratarata << endl;      return 0;  } |

|  |
| --- |
| Menghitung Nilai Rata-rata Mahasiswa  Nama : Agus  Jumlah Nilai : 3  Nilai Ke-1 : 60  Nilai Ke-2 : 70  Nilai Ke-3 : 80  Nilai Rata-rata adalah : 70  === Code Execution Successful === |

1. Buat elemen array sebanyak 10 buah, masukkan nilai 8 pada array yang berbeda, kemudian buat program untuk menampilkan posisi array yang bernilai 8 tersebut.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {    int arr[10] = {0};  arr[3] = 8;  arr[2] = 8;  arr[9] = 8;  arr[8] = 8;  arr[5] = 8;    cout << "Posisi (indeks) array yang bernilai 8:" <<endl;  for (int i = 0; i < 10; i++) {  if (arr[i] == 8) {  cout << "Array ke-" << i << endl;  }  }  return 0;  } |

|  |
| --- |
| Posisi (indeks) array yang bernilai 8:  Array ke-2  Array ke-3  Array ke-5  Array ke-8  Array ke-9  === Code Execution Successful === |

1. Buat matriks 2 dimensi, inputkan datanya kemudian tambahkan dan kalikan dengan bilangan 5.
2. Penjumlahan

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {      int matriksA[2][2] = {{1, 2}, {3, 4}};      int matriksB[2][2] = {{5, 6}, {7, 8}};      int hasil[2][2];        for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              hasil[i][j] = matriksA[i][j] + matriksB[i][j];          }      }      cout << "Hasil penjumlahan matriks: "<<endl;      for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              cout << hasil[i][j] << endl;          }      }      return 0;  } |

|  |
| --- |
| Hasil penjumlahan matriks:  6  8  10  12  === Code Execution Successful === |

1. Perkalian

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {      int matriksA[2][2] = {{1, 2}, {3, 4}};      int matriksB[2][2] = {{5, 6}, {7, 8}};      int hasilA[2][2];      int hasilB[2][2];        for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              hasilA[i][j] = matriksA[i][j]\*5;          }      }        for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              hasilB[i][j] = matriksB[i][j]\*5;          }      }      cout << "Hasil perkalian matriksA: "<<endl;      for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              cout << hasilA[i][j] << endl;          }      }        cout << "Hasil perkalian matriksB: "<<endl;      for (int i = 0; i < 2; i++) {          for (int j = 0; j < 2; j++) {              cout << hasilB[i][j] << endl;          }      }      return 0;  } |

|  |
| --- |
| Hasil perkalian matriksA:  5  10  15  20  Hasil perkalian matriksB:  25  30  35  40  === Code Execution Successful === |

1. Kesimpulan

Array adalah struktur data yang menyimpan elemen dengan tipe data yang sama secara berurutan. A C++ array memiliki ukuran tetap.

[Array](https://www.guru99.com/id/cpp-dynamic-array.html) sangat penting dalam bahasa pemrograman apa pun. Mereka menyediakan cara yang lebih nyaman untuk menyimpan variabel atau kumpulan data dengan tipe data serupa secara bersamaan daripada menyimpannya secara terpisah. Setiap nilai array akan diakses secara terpisah.

1. Buat program untuk mencari nilai rata-rata seorang mahasiswa
2. Buat program untuk mencari nilai rata-rata seorang mahasiswa.
3. Buat program untuk mencari nilai rata-rata seorang mahasiswa.
4. Deklarasikan sebuah array bertipe char dengan panjang 20. Kemudian inputkan sebuah karakter dari keyboard dan cari apakah karakter tersebut ada dalam array yang telah dideklarasikan
5. Deklarasikan sebuah array bertipe char dengan panjang 20. Kemudian inputkan sebuah karakter dari keyboard dan cari apakah karakter tersebut ada dalam array yang telah dideklarasika