

## PERTEMUAN 12 ARRAY SATU DIMENSI

### A. Tujuan Pembelajaran

Setelah mempelajari materi ini, mahasiswa mampu :

1. Memahami konsep array 1 dimensi
2. Merekayasa array 1 dimensi dalam Bahasa pemrograman
3. Menggunakan array 1 dimensi dalam Bahasa pemrograman

### B. Uraian Materi

Array adalah sekelompok lokasi memori yang berdekatan yang semuanya memiliki tipe data yang sama. Untuk merujuk pada suatu lokasi atau elemen tertentu dalam array, kita bisa menentukan nama, jumlah elemen dan posisi array tertentu didalamnya.(Edition n.d.)

Ada beberapa sifat yang dimiliki oleh sebuah array :

1. Sekelompok lokasi memori (data) yang mempunyai tipe data yang sama dan menggunakan nama yang sama
2. Sekelompok variabel dapat menggunakan nama yang sama
3. Perbedaan antara variabel dengan variable yang lain adalah apa yang terkandung didalam subscript.
4. Subscript adalah bilangan yang terdapat di dalam kurung kotak [ ]
5. Cara mengakses sebuah array adalah dengan memanggil variable yang ada dalam subscript

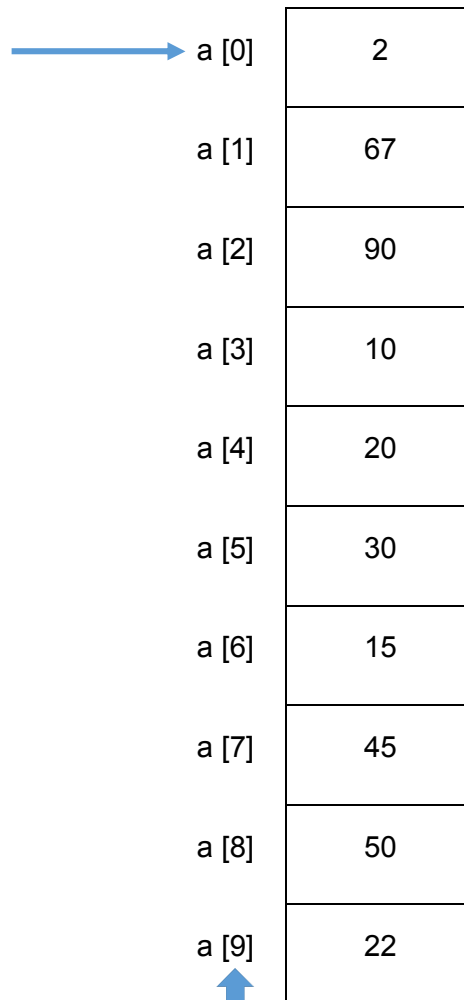
#### 1.1 Array Satu Dimensi

Seperti yang sudah dikatakan diatas, bahwa elemen – elemen diatas mempunyai susunan tertentu. Susunan tersebut bisa berupa satu dimensi, dua dimensi bahkan N dimensi. Tentu susunan yang termudah adalah satu dimensi atau berdimensi satu, sehingga biasa disebut array satu dimensi atau array dimensi satu (one dimension array).

Gambar dibawah ini menunjukkan bahwa array tersebut mempunyai nama a, yang mengandung 10 elemen. Elemen – elemen ini diberikan nama berdasarkan letak atau posisi nomor elemen yang ditandai dengan kurung kotak ( [ ] ). Array

pertama didalam urutan atau tata letak, selalu dimulai dari urutan ke Nol ( 0 ). Dalam pemberian nama pada sebuah array bisa menggunakan huruf-huruf, angka-angka dan garis bawah tetapi tidak bisa menggunakan hanya 1 angka saja.

Bisa dilihat, semua dalam array mempunyai nama yang sama yaitu a.



|       |    |
|-------|----|
| a [0] | 2  |
| a [1] | 67 |
| a [2] | 90 |
| a [3] | 10 |
| a [4] | 20 |
| a [5] | 30 |
| a [6] | 15 |
| a [7] | 45 |
| a [8] | 50 |
| a [9] | 22 |

Posisi penomoran dalam array a

Array satu dimensi atau biasa disebut *vector* (karena mempunyai satu arah) dapat digunakan untuk berbagai macam keperluan. Kadang – kadang penggambaran array satu dimensi dapat digambarkan dalam contoh dibawah ini.

|   |  |  |
|---|--|--|
| 0 |  | Biasanya digunakan untuk menggambarkan suatu daftar atau list, sehingga array satu dimensi ini sering disebut dengan <b>LIST</b> |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |

|   |  |  |
|---|--|--|
| 6 |  | Array satu dimensi yang digambarkan seperti ini biasanya untuk mengilustrasikan struktur data yang bersifat tumpukan atau <b>STACK</b> |
| 5 |  |  |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 |  |  |

## 1.2 Deklarasi Array

Untuk mendeklarasikan sebuah array satu dimensi dalam sebuah Bahasa pemrograman biasanya digunakan tanda [ ] (Kurung Kotak) / *Bracket* yang berfungsi untuk menyatakan berapa banyak jumlah elemen yang akan terbentuk pada sebuah array tersebut.

Adapun bentuk umum dari pendeklarasian array satu dimensi :

```
Tipe_data <spasi> Nama_array [Jumlah_Elemen];
```

Contoh :

```
Int Nilai[5]
```

Pada contoh yang diberikan diatas dinyatakan bahwa array tersebut mempunyai :

Tipe data Array : Integer

Nama Array : Nilai

Jumlah elemen Array : 5

Atau bisa juga pendeklarasian array satu dimensi ini langsung kita deklarasikan disertai dengan isi atau inisialisasi untuk array tersebut.

```
Tipe_data <spasi> Nama_array [Jumlah_Elemen] = {elemen 1, elemen 2, ....., elemen ke N};
```

Contoh :

```
Int Nilai[5]={70,80,100,50,10};
```

Pada contohyang diberikan diatas dinyatakan bahwa array tersebut mempunyai :  
Tipe data Array : Integer

Nama Array : Nilai

Jumlah elemen Array : 5

Nilai Elemen Array : [1] = 70

[2] = 80

[3] = 100

[4] = 50

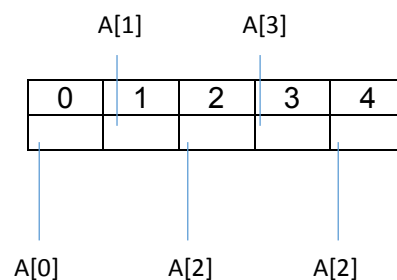
[5] = 10

### 1.3 Menyiapkan Array satu dimensi pada Bahasa pemrograman C++

Cara menyiapkan array satu dimensi adalah dengan menyebut tipe\_data, Nama\_array dan jumlah\_element array seperti contoh dibawah ini.

```
#include<iostream.h>
Void main ()
{
  Int A[5] ;
```

Dengan perintah A[5]; maka akan disiapkan sebuah array satu dimensi dengan tipe integer yang memuat 5 elemen yang akan diberi nomor indeks dari 0 sampai 4 yang dapat diilustrasikan dengan gambar berikut :



Karena ada 5 elemen, maka masing-masing elemen akan diberikan nama atau sebutan yang berbeda dengan menggunakan nomor indeks, sehingga masing-masing menjadi: A[0], A[1], A[2], A[3], A[4] yang bisa dibaca dengan :

A dengan indeks 0

A dengan indeks 1

A dengan indeks 2

A dengan indeks 3

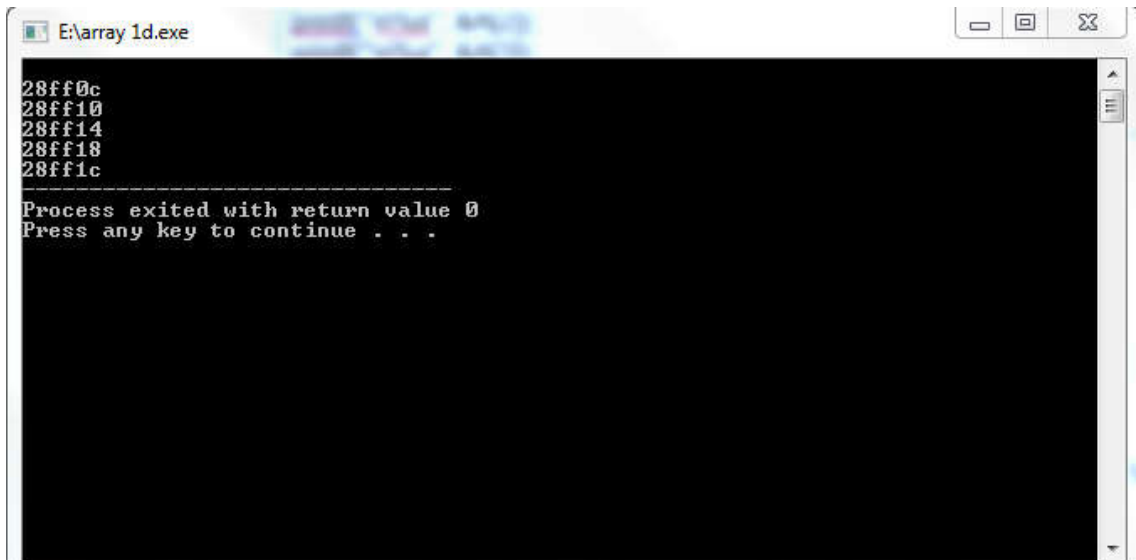
A dengan indeks 4

### 1.4 Alamat elemen – elemen Array satu dimensi

Untuk mengetahui besarnya alamat-alamat yang dimiliki oleh setiap elemen array, maka bisa kita gunakan perintah seperti dibawah ini :

```
#include <stdio.h>
main ()
{
int A[5];
printf("\n%x", &A[0]);
printf("\n%x", &A[1]);
printf("\n%x", &A[2]);
printf("\n%x", &A[3]);
printf("\n%x", &A[4]);
}
```

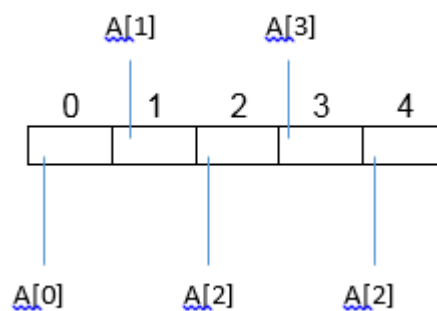
Maka akan tercetak :



```
E:\array 1d.exe
28ff0c
28ff10
28ff14
28ff18
28ff1c
-----
Process exited with return value 0
Press any key to continue . . .
```

Penjelasan :

Dengan dituliskannya int A[5] maka akan terbentuk ilustrasi sebagai berikut :



Tanda & digunakan untuk menunjukan posisi atau alamat. &A[0] berarti posisi atau alamat elemen A[0].

Untuk mencetak posisi atau alamat suatu area, dapat menggunakan perintah %x.

Alamat yang tercetak adalah alamat elemen dalam notasi bilangan Hexadesimal. Sifat dari alamat yang tercetak ini berbeda beda, tergantung dari computer yang digunakan.

Pada contoh yang diberikan diatas, array yang bertipe integer akan mebuat elemen yang mempunyai panjang 4 byte per elemen, sehingga alamat tiap-tiap elemenselalu berselisih 4 :

Alamat A[1] merupakan sambungan dari alamat A[0]

atau

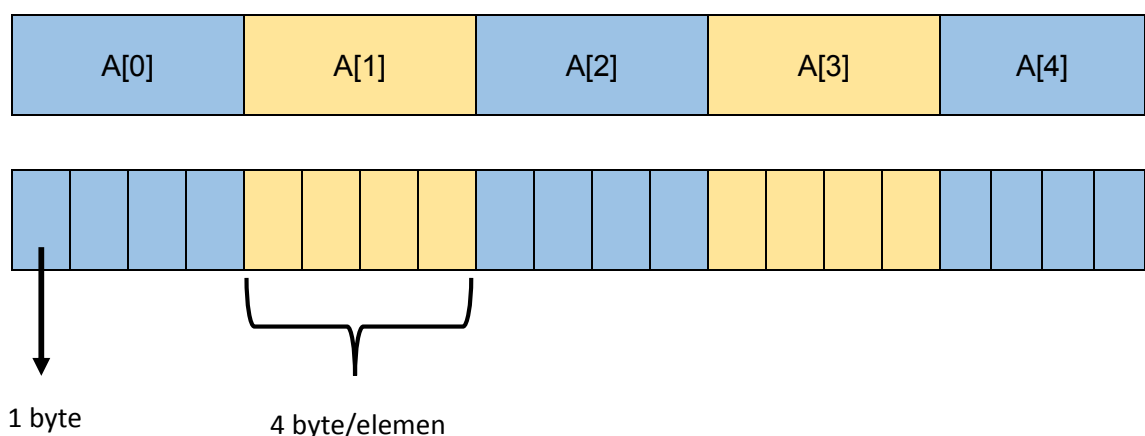
Alamat A[1] = alamat A[0] + 4

Alamat A[2] = alamat A[1] + 4

Alamat A[3] = alamat A[2] + 4

Alamat A[4] = alamat A[3] + 4

Contoh tersebut juga dapat diilustrasikan seperti gambar berikut



Jumlah elemen array yang ada adalah = 5 elemen

Panjang array = 5 elemen \* 4 byte/elemen = 20 byte

**Contoh perhitungan array satu dimensi**

1. Suatu array dideklarasikan dengan A[5], array tersebut mempunyai tipe data integer sehingga masing-masing elemen terdiri dari 4 byte. Jika 28ff0c<sub>(H)</sub> adalah alamat dari elemen pertama, tentukan :
  - a) Jumlah elemen yang terdapat dalam array satu dimensi tersebut
  - b) Panjang array satu dimensi tersebut dalam satuan byte
  - c) Alamat dari elemen A[4]
  
2. Suatu array dideklarasikan dengan A[9], array tersebut mempunyai tipe data long integer sehingga masing-masing elemen terdiri dari 8 byte. Jika 16FF<sub>(H)</sub> adalah alamat dari elemen pertama, tentukan :
  - a) Jumlah elemen yang terdapat dalam array satu dimensi tersebut
  - b) Panjang array satu dimensi tersebut dalam satuan byte
  - c) Alamat dari elemen A[5]

**Penyelesaian**

1. Alamat dari elemen A[0] = 28ff0c<sub>(H)</sub>  
 Masing-masing elemen terdiri dari = 4 byte/elemen
  - a) Jumlah elemen yang terdapat dalam array satu dimensi tersebut = 5 elemen
  - b) Panjang array satu dimensi tersebut dalam satuan byte = 5 \* 4 byte/elemen  
 = 20 byte
  - c) Alamat dari elemen A[4] = ?  
 Pergeseran = (4-0) elemen \* 4 byte/elemen  
 = 16<sub>(10)</sub> = 10<sub>(H)</sub>  
 &A[4] = 28ff0c<sub>(H)</sub> + 10<sub>(H)</sub> = 28ff1c<sub>(H)</sub>
  
2. Alamat dari elemen A[0] = 16FF<sub>(H)</sub>  
 Masing-masing elemen terdiri dari = 8 byte/elemen
  - a) Jumlah elemen yang terdapat dalam array satu dimensi tersebut = 9 elemen
  - b) Panjang array satu dimensi tersebut dalam satuan byte = 9 \* 8 byte/elemen  
 = 72 byte
  - c) Alamat A[5] = ?  
 Pergeseran = (5-0) elemen \* 8 byte/elemen  
 = 40<sub>(10)</sub> = 28<sub>(H)</sub>  
 &A[4] = 16FF<sub>(H)</sub> + 28<sub>(H)</sub> = 1727<sub>(H)</sub>

### 1.5 Membuat Array Numerik Dengan Nilai Awal


Dalam membuat Array Numerik dengan nilai awal, bisa dilakukan dengan cara mendeklarasikan jumlah elemen array terlebih dahulu maupun tidak dideklarasikan.

Dalam memberikan nilai dalam sebuah elemen array dapat dilakukan dengan cara menuliskan angka-angka yang akan dimuat dalam elemen array dengan dihaluai dan ditutup dengan tanda kurung kurawal ( { } ) dan antar angka yang satu dengan angka yang lain dipisahkan dengan tanda koma.

Contoh

```
#include <stdio.h>
main ()
{
int l;
int A[5] = {12,13,14,15,16};
for (l=0; l<=4; l++)
{
printf ("%3i",A[l]);
}
}
```

Maka akan tercetak :



Penjelasan :

Dengan source code tersebut disiapkan array dengan 5 (Lima) elemen dengan isinya yang ada didalam tanda kurung kurawal seperti ilustrasi gambar dibawah ini :

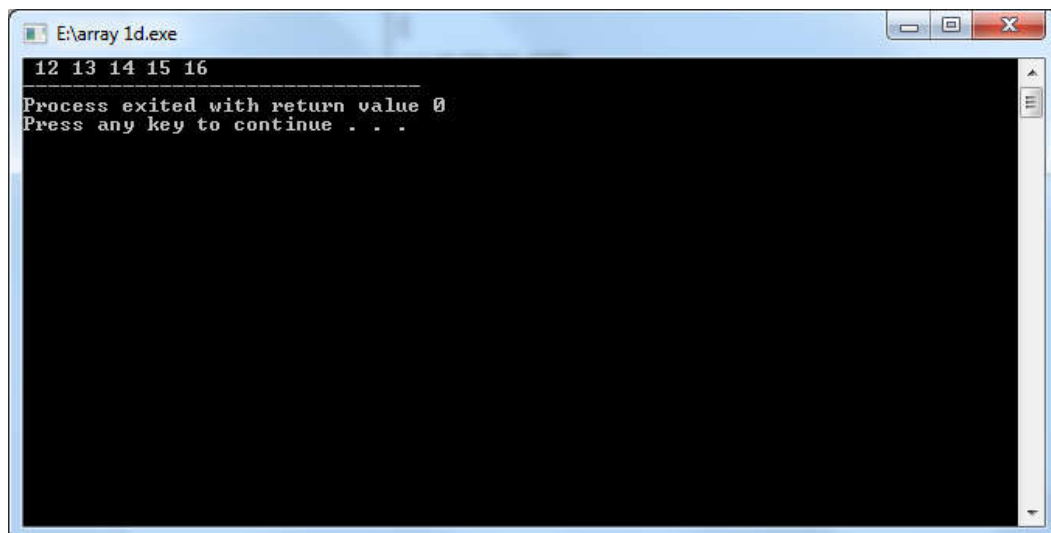


| 0    | 1    | 2    | 3    | 4    |
|------|------|------|------|------|
| 12   | 13   | 14   | 15   | 16   |
| A[0] | A[1] | A[2] | A[3] | A[4] |

Ada berbagai cara untuk menyiapkan array satu dimensi yang mengandung isi elemen didalamnya. Seperti contoh dibawah ini

```
#include <stdio.h>
main ()
{
    int i;
    int A[ ] = {12,13,14,15,16};
    for (i=0; i<=4; i++)
    {
        printf ("%3i",A[i]);
    }
}
```

Maka akan tercetak




Bisa dilihat pada contoh yang diberikan diatas pendeklarasian array satu dimensi dengan tipe data integer dan nama array A tidak disnutkan berapa banyak ruamh atau elemen yang harus disiapkan oleh program tersebut. Tetapi karna setelahnya di deklarasikan data yang akan mengisi elemen elemen yang ada dalam array satu dimensi tersebut maka ruangan yang disiapkan adalah sesuai data yang akan

dimasukan kedalam elemen-elemen array tersebut. Dalam contoh ada lima data yang akan dimasukan sehingga disiapkanlah lima ruangan atau elemen.

Pembentukan elemen – elemen array satu dimensi juga bisa menggunakan perintah `#define` di awal penulisan program atau header program. Seperti contoh berikut :

```
#include <stdio.h>
#define n 5
main ()
{
    int i;
    int A[n] = {1,2,90,98,55};
    for (i=0; i<=4; i++)
    {
        printf ("%3i",A[i]);
    }
}
```

Maka akan tercetak



Penjelasan :

n seperti contoh diatas didefinisikan sebagai lima (5) sehingga n tidak dapat diubah dan data yang akan dimasukan kedalam elemen array pun tidak dapat melebihi lima data. Jika terjadi kelebihan dalam memasukan data maka akan ditampilkan pesan error karena ruangnya yang disediakan tidak cukup untuk menampung data yang akan dimasukan.

**C. Soal Latihan / Tugas**

1. Array harus memiliki tipe data yang . . . .
2. Pendeklarasian elemen pada array satu dimensi biasanya menggunakan tanda baca ....
3. Pada Array satu dimensi, setiap elemen bertipe data integer memiliki jumlah byte.....dan pada long integer berjumlah.....byte
4. Suatu array satu dimensi dideklarasikan dengan  $A[10]$ , berapa panjang array dalam byte yang ada pada array tersebut jika tipe data yang digunakan adalah long integer....
5. Suatu array mempunyai alamat pertama  $A[0] = 18FF2_{(H)}$ , jika array tersebut bertipe data integer tentukan alamat dari elemen  $A[7]$
6. Buatlah source code atau program sederhana untuk memasukkan dan menampilkan array satu dimensi dengan 10 elemen yang didalamnya memuat data 80, 76, 87, 10, 67, 90, 77, 54, 9, 32

**D. Referensi**

Data, R. (t.thn.). *Data, Refsnes*. Dipetik October 22, 2019, dari [https://www.w3schools.com/cpp/cpp\\_booleans.asp](https://www.w3schools.com/cpp/cpp_booleans.asp)

Davis, S. R. (2014). *C++ For Dummies* (7th ed.). John Wiley & Sons, Inc.

Deitel, P., & Deitel, H. (2014). *C++ How To Program* (9th ed.). United State of America: Pearson.

Halterman, R. L. (2019). *Fundamental of C++ Programming*.

Kirch-Prinz, U., & Kirch-Prinz, U. (2002). *A Complete Guide to Programming in C++*. Sudbury: Jones and Bartlett Publishers.

S, R. A. (2018). *Logika Algoritma dan Pemrograman Dasar*. Bandung: Modula.

Sjukani, M. (2014). *Algoritma dan Struktur Data 1 dengan C, C++ dan Java* (Edisi 9 ed.). Jakarta: Mitra Wacana Media.