

MODUL MATA KULIAH

ANALISIS DAN DESAIN ALGORITMA

PG167 – 3 SKS



**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BUDI LUHUR**

**JAKARTA
SEPTEMBER 2019**

TIM PENYUSUN

Ita Novita, S.Kom., M.T.I
Dr. Achmad Solichin, S.Kom., M.T.I
Atik Ariesta, S.Kom., M.Kom



MODUL PERKULIAHAN #9

STRUKTUR ARRAY SATU DIMENSI

Capaian Pembelajaran	:	Mahasiswa mampu memahami penggunaan variabel struktur array satu dimensi
Sub Pokok Bahasan	:	<ul style="list-style-type: none">1.1. Konsep Array Satu Dimensi1.2. Contoh algoritma operasi dasar array1.3. Contoh algoritma yang melibatkan array satu dimensi secara sederhana
Daftar Pustaka	:	<ul style="list-style-type: none">1. Gaddis, nd.2011. Starting Out with C++ from Control Structures through Objects .8th. Boston: Addison-Wesley.2. Institue of Distance & Open Learning, n.d. UNIT I Algorithms, Flowcharts & Program Design in: INTRODUCTION TO C++. p. 2053. Sjukani,Moh .2014. Algoritma (Algoritma & Struktur Data 1) Dengan C, C++, dan Java Edisi 9", Mitra Wacana Media.

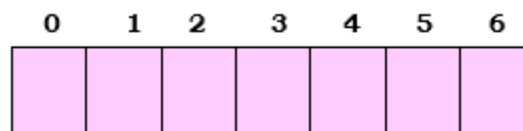
STRUKTUR ARRAY SATU DIMENSI

1.1. KONSEP ARRAY SATU DIMENSI

PENGERTIAN ARRAY SATU DIMENSI

Array dapat diartikan sebagai sesuatu yang berbaris atau berderet-deret. Dalam bahasa pemrograman, array adalah variabel sejenis yang berderet-deret sedemikian rupa sehingga alamatnya saling bersambung atau bersebelahan/berdampingan (contiguous).

Karena variabel saling bersambung, maka array satu dimensi biasanya diilustrasikan dengan gambar sebagai berikut:



Dari ilustrasi di atas, terlihat sebuah array satu dimensi, yang digambarkan dengan 7 buah kotak. Yang disebut kotak disini, dalam istilah array disebut dengan elemen, cell (sel), lokasi atau kolom. Jadi array diatas terdiri dari 7 elemen, atau 7 lokasi, atau 7 kolom. Untuk bahasa C/C++ dan Java, elemen pertama diberi nomor 0, yang dilanjutkan dengan nomor 1,2 dan seterusnya.

MENYIAPKAN ARRAY SATU DIMENSI

Menyiapkan array satu dimensi dilakukan dengan cara menyebutkan tipe, nama dan jumlah elemen.

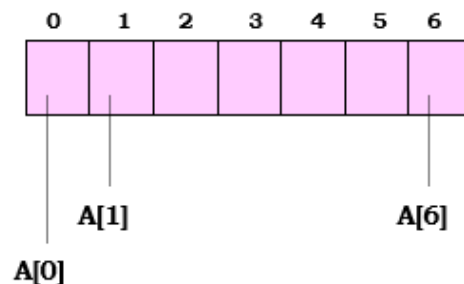
Contoh:

C

```
#include<stdio.h>
void main()
{ int A[7];
```



Dengan instruksi `int A[7]`, akan disiapkan sebuah array satu dimensi bertipe `int` dengan 7 elemen yang diberi nomor indeks (index) dari 0 sampai dengan 6 yang diilustrasikan dengan gambar sebagai berikut:



A =variabel array

Karena ada 7 elemen, maka setiap elemen diberi sebutan nama yang berbeda dengan memberikan nomor indeks (index), sehingga masing-masing menjadi: `A[0]`, `A[1]`, `A[2]`, sampai dengan `A[6]`, yang biasa disebut (dibaca) dengan :

A dengan indeks 0

A dengan indeks 1

Dan seterusnya

ALAMAT ELEMEN-ELEMEN ARRAY SATU DIMENSI

Alamat elemen suatu array saling bersambung/bersebelahan (contiguous).

Contoh:

Contoh-1. dengan C

```
#include<stdio.h>
void 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] );
}
```

Akan tercetak :

21E6
21E8
21EA
21EC
21EE

format : **%X**



integer : 2 Byte

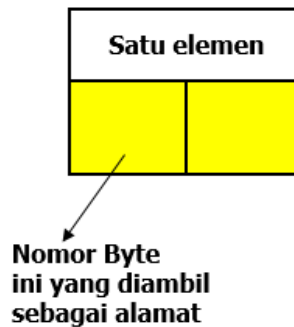


Alamat dicetak dengan notasi bilangan Hexadecimal.
 Alamat ini bisa berbeda bila dirun pada komputer(sistem) yang berbeda.



Tanda & artinya alamat, &A[0] artinya alamat elemen A[0]. Sedangkan yang dimaksud dengan alamat adalah nomor byte pertama (diilustrasikan dengan byte paling kiri) dari suatu elemen sebagai berikut:

Satu elemen
integer = 2 byte



1.2. CONTOH ALGORITMA OPERASI DASAR ARRAY

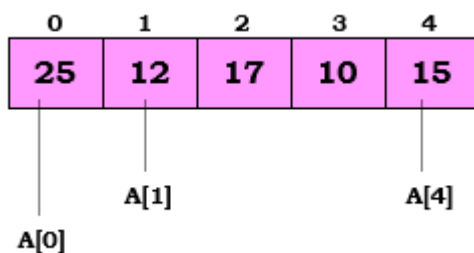
MENYIAPKAN ARRAY NUMERIK SATU DIMENSI LENGKAP DENGAN ISINYA

Contoh:

Algoritma:

Inisialisasi variabel Array $A[5] = \{ 25, 12, 17, 10, 15 \}$

Ilustrasi:



Penulisan dalam Bahasa C (Cara 1)

```
#include <stdio.h>
void main()
{
    int A[5] = { 25,12,17,10,15 };
    -
    -
}
```

Penulisan dalam Bahasa C (Cara 2)

```
#include <stdio.h>
void main()
{
    int A[5] = { 25,12,17,
                10,15 };
    -
    -
}
```

Penulisan dalam Bahasa C (Cara 3)

```
#include <stdio.h>
void main()
{
    int A[ ] = { 25,12,17,10,15 };
    -
    -
}
```

Penulisan dalam Bahasa C (Cara 4)

```
#include <stdio.h>
#define n 5
void main()
{
    int A[n] = { 25,12,17,10,15 };
    -
    -
}
```



Catatan:

1. Jika jumlah data yang akan diisi lebih banyak dari jumlah elemen yang disediakan, maka akan terjadi error sewaktu program dicompile.

Contoh:

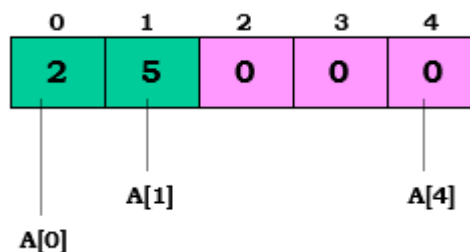
```
#include <stdio.h>
void main()
{
    int A[5] = { 25,12,17,10,15, 19 };
    -
}
```

Akan terjadi error, karena jumlah data yang disimpan 6 buah, sedangkan jumlah elemen yang disediakan hanya 5 elemen.

2. Untuk array numerik, bila diisi hanya sebagian elemen (mulai dari elemen pertama), maka sisa elemen yang tidak diisi, oleh Bahasa C otomatis diisi dengan nol.

Contoh:

```
#include <stdio.h>
void main()
{
    int A[5] = { 2, 5 };
    -
}
```

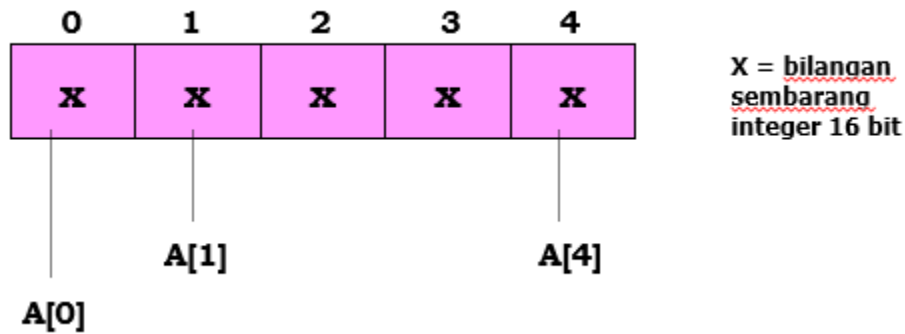


3. Bila menyiapkan array tanpa dengan isinya, maka lokasi array isinya tetap seperti semula, apa adanya waktu itu. Sehingga isinya tidak diketahui.

Contoh:

```
#include <stdio.h>
void main()
{
    int A[5];
    -
}
```





MENYIAPKAN ARRAY CHARACTER SATU DIMENSI LENGKAP DENGAN ISINYA

Contoh:

Algoritma:

Inisialisasi variabel Array C[5]="ABCED"

Ilustrasi:

0	1	2	3	4
A	B	C	D	E

Penulisan dalam Bahasa C (Cara 1)

```
#include <stdio.h>
void main()
{
    char C[5] = "ABCDE";
    -
    -
}
```

Penulisan dalam Bahasa C (Cara 2)

```
#include <stdio.h>
void main()
{
    char C[5] = { 'A', 'B', 'C', 'D', 'E' };
    -
    -
}
```



Catatan:

1. Bila diisi hanya sebagian elemen (mulai dari elemen pertama), maka sisa elemen yang tidak diisi, akan diisi dengan karakter NULL

Contoh:

```
#include <stdio.h>
void main()
{
    char C[5] = "ABC";
    -
    -
}
```

0	1	2	3	4
A	B	C	\0	\0

character
NULL

Character NULL

0 0 0 0 0 0 0 0

1 Byte = 8 bit
Semua bitnya OFF

Bila dicetak dengan:

```
for(I=0; I <= 4; I++)
{ printf ("%c", C[ I ] ); }
printf("Selesai");
```

Tercetak :ABC■■Selesai

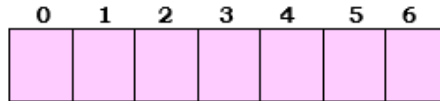
↑
Dua karakter NULL



MENGISI ARRAY SATU DIMENSI DENGAN NILAI NUMERIK

Contoh 1:

Sudah ada array integer satu dimensi yang dibuat dengan `int A[7]`, belum ada isinya dengan ilustrasi sebagai berikut:



Buatlah algoritma dan flowchart untuk mengisi array A sehingga hasilnya sebagai berikut:



Jawab:

Algoritma:

Deklarasi variabel array `A[7]`, `I`

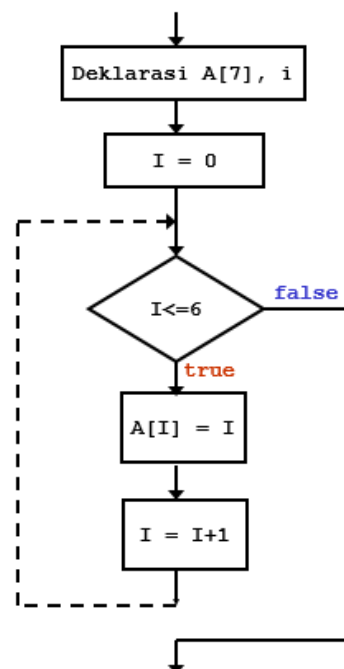
`I = 0`

Lakukan selama `I <= 6`

`A[I] = I`

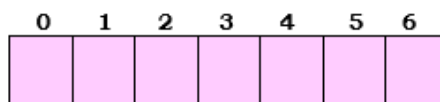
`I = I + 1`

Flowchart:



Contoh 2:

Sudah ada array integer satu dimensi yang dibuat dengan `int A[7]`, belum ada isinya dengan ilustrasi sebagai berikut:



Buatlah algoritma dan flowchart untuk mengisi array A sehingga hasilnya sebagai berikut:

0	1	2	3	4	5	6
	1		3		5	

Jawab:

Algoritma:

Deklarasi variabel array A[7], I

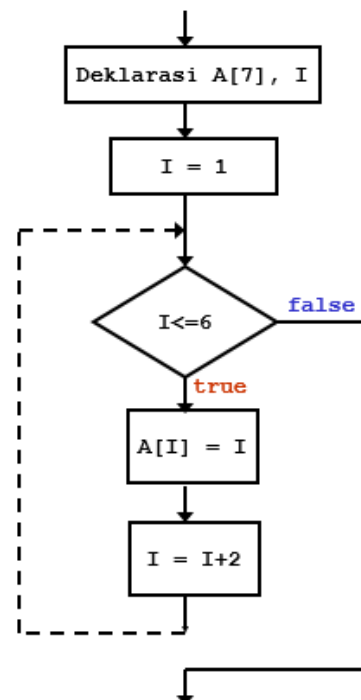
I = 1

Lakukan selama I ≤ 6

A[I] = I

I = I + 2

Flowchart:



MENGISI ARRAY DENGAN DATA YANG DIKETIK MELALUI KEYBOARD

Contoh 1:

Dalam lembar dokumen ada data berupa nilai ujian mahasiswa sebagai berikut:

57, 75, 90, 55, 60, 62, 72, 58, 76, 69, 67, 82, 65, 48, 79, 64, 50, dst.

Sudah ada (sudah dibuat) array satu dimensi int A[11] belum ada isinya. Susun algoritma untuk menginput data dari luar melalui keyboard, dan mengisinya ke dalam array sehingga isi array menjadi:

0	1	2	3	4	5	6	7	8	9	10
57	75	90	55	60	62	72	58	76	69	67



Jawab:

Algoritma:

Deklarasi variabel array A[11], I, X

I = 0

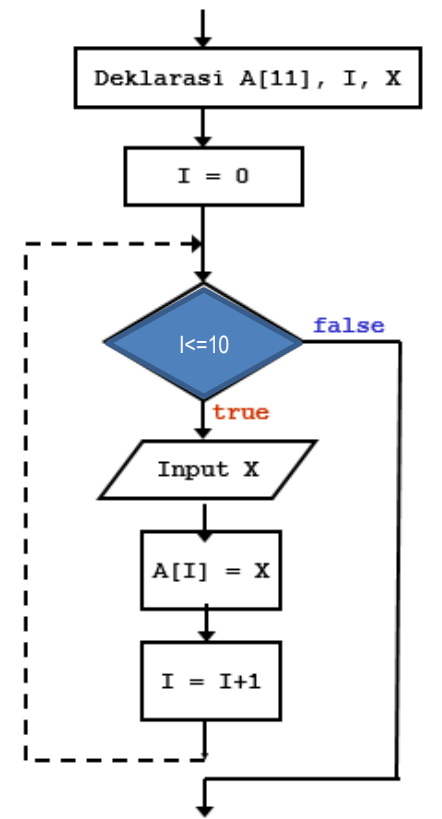
Lakukan selama I ≤ 10

Input X

A[I] = X

I = I + 1

Flowchart:



Program 9.1 Program Mengisi Array Melalui Keyboard Dalam Bahasa C

```
#include<stdio.h>
void main()
{ int A[[11], X, I;

    for(I=0; I<=10; I++)
    { scanf("%i", &X );
      A[ I ] = X;
    }
}
```



Contoh 2:

Dalam lembar dokumen ada data berupa nilai ujian mahasiswa sebagai berikut:

57, 75, 90, 55, 60, 62, 72, 58, 76, 69, 67, 82, 65, 48, 79, 64, 50, dst.

Sudah ada (sudah dibuat) array satu dimensi int A[11] belum ada isinya. Susun algoritma untuk menginput data dari luar melalui keyboard. Bila data yang diinput nilainya < 60 , tidak disimpan kedalam array, tetapi bila nilai yang diinput ≥ 60 , maka akan disimpan ke dalam array sehingga isi array menjadi:

0	1	2	3	4	5	6	7	8	9	10
75	90	60	62	72	76	69	67	82	65	79

Jawab:

Algoritma:

Deklarasi variabel array A[11], I, X

I = 0

Lakukan selama I ≤ 10

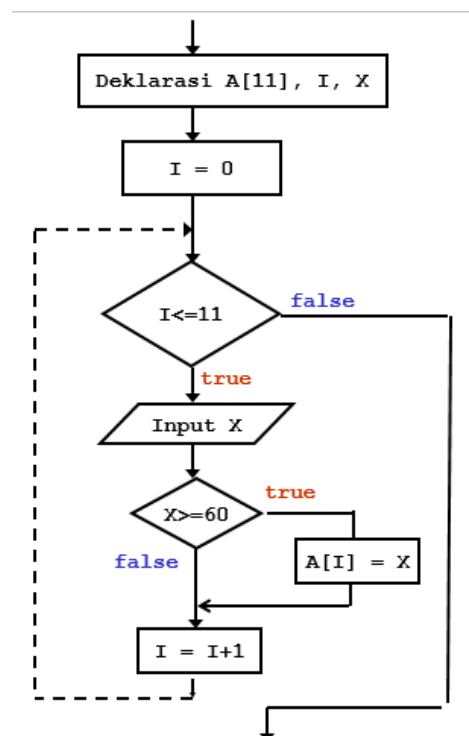
 Input X

 Cek apakah X ≥ 60

 Jika benar maka A[I] = X

 I = I + 1

Flowchart:



Program 9.2 Program Mengisi Array Melalui Keyboard Untuk Isi Array ≥ 60

```
#include<stdio.h>
void main()
{ int A[[11], X, I;

  for(I=0; I<=10; I++)
  { scanf("%i", &X );
    If (X>=60)
      A[ I ] = X;
  }
}
```

MENCETAK ISI ARRAY SATU DIMENSI

Contoh 1:

Sudah ada array satu dimensi A[11], sudah ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10
12	17	10	5	15	25	11	8	3	16	19

Susun algoritma dan flowchart untuk mencetak isi array sehingga tercetak dengan urutan sebagai berikut:

12 17 10 5 15 25 11 8 3 16 19

Jawab:

Algoritma:

Deklarasi variabel array A[11], I

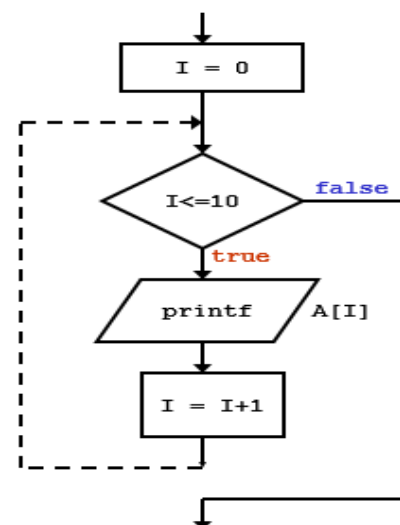
I= 0

Lakukan selama I<=10

 Cetak A[I]

 I=I+1

Flowchart:



Contoh 2:

Sudah ada array satu dimensi A[11], sudah ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10
12	17	10	5	15	25	11	8	3	16	19

Susun algoritma dan flowchart untuk mencetak isi array sehingga tercetak dengan urutan sebagai berikut:

19 16 3 8 11 25 15 5 10 17 12

Jawab:

Algoritma:

Deklarasi variabel array A[11], I

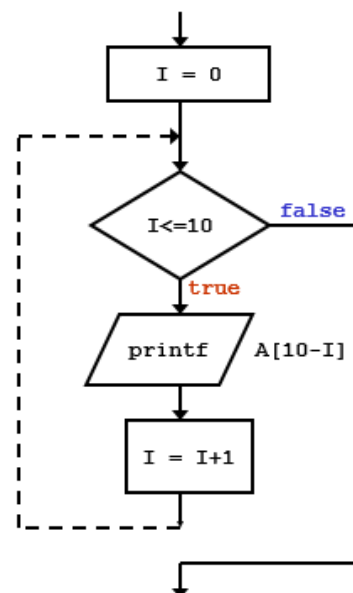
I = 0

Lakukan selama I <= 10

 Cetak A[10-I]

 I = I + 1

Flowchart:



Contoh 3:

Sudah ada array satu dimensi A[11], sudah ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10
12	17	10	5	15	25	11	8	3	16	19

Susun algoritma dan flowchart untuk mencetak isi array yang nilainya **lebih besar dari 10** sehingga tercetak dengan urutan sebagai berikut:

12 17 15 25 11 16 19



Jawab:

Algoritma:

Deklarasi variabel array A[11], I

I = 0

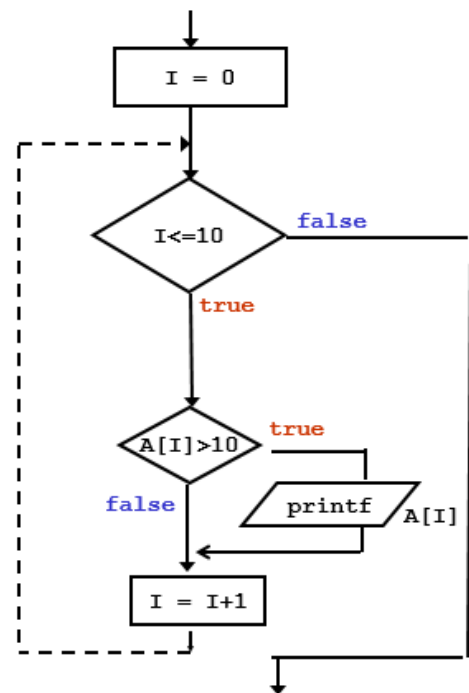
Lakukan selama I ≤ 10

 Cek apakah A[I] > 10

 Jika benar maka Cetak A[I]

 I = I + 1

Flowchart:



Program 9.3 Program Mencetak Isi Array yang Nilainya > 10

```
#include<stdio.h>
void main()
{ int A[[11]={12,17,10,5,15,25,11,8,3,16,19}, I;

  for(I=0; I<=10; I++)
  { if (A[I]>10)
    printf("%3i", A[I] );
  }
}
```



1.3. CONTOH ALGORITMA YANG MELIBATKAN ARRAY SATU DIMENSI SECARA SEDERHANA

Contoh 1:

Sudah ada array satu dimensi yang dibuat dengan int A[11]. Sudah ada isinya nilai-nilai numerik. Susun algoritma untuk mencetak Total isi array tersebut.

0	1	2	3	4	5	6	7	8	9	10
12	17	10	5	15	25	11	8	3	16	19

Jawab:

Algoritma:

Deklarasi variabel array A[11]={12,17,10,5,15,25,11,8,3,16,19}, T, I

T=0

I=0

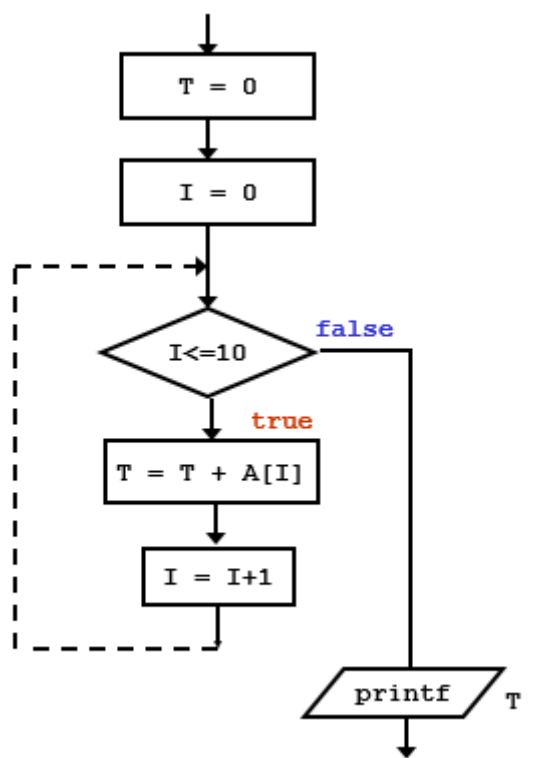
Lakukan selama I<=10

 T = T + A[I]

I=I+1

Cetak T

Flowchart:



Contoh 2:

Sudah ada array satu dimensi yang dibuat dengan char A[15]. Sudah ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

Susun algoritma untuk mencetak isi array tersebut tercetak dengan urutan sebagai berikut:

```
A B C D E
F G H I J
K L M N O
```

Jawab:**Algoritma:**

Deklarasi variabel array C[15]={‘A’,‘B’,‘C’,‘D’,‘E’,‘F’,‘G’,‘H’,‘I’,‘J’,‘K’,‘L’,‘M’,‘N’,‘O’},
N, I, J

N=0

I=1

Lakukan selama I<=3

 J=1

 Lakukan selama J<=5

 Cetak array A[N]

 N++

 J++

I++



Program 9.4 Program Mencetak Isi Array Karakter

```
#include<stdio.h>
void main()
{ char C[15]={ 'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O'}, I, J, N;
  N = 0;
  for(I=1; I <= 3; I++)
    { for(J=1; J<=5; J++)
      { printf("%2c", A[ N ] );
        N = N + 1;
      }
      printf("\n");
    }
}
```



SOAL LATIHAN

1. Sudah ada array A satu dimensi yang dibuat dengan int A[9]. Belum ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8

Gambarkan kembali array tersebut lengkap dengan isinya bila diisi dengan penggalan program sebagai berikut:

a.

```
I = 0;
while (I<=8)
{ A[I] = I
  I+=2;
}
```

b.

```
For (I=0;I<=8;I++)
{ A[I] = I*2+1
```

2. Sudah ada array A satu dimensi yang dibuat dengan int A[11]. Belum ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10

Susun algoritma untuk mengisi array tersebut sehingga isinya menjadi sebagai berikut:

a.

0	1	2	3	4	5	6	7	8	9	10
1		2		3		4		5		6

b.

0	1	2	3	4	5	6	7	8	9	10
1		3		5		7		9		11

3. Sudah ada array A satu dimensi yang dibuat dengan int A[11]. Belum ada isinya dengan ilustrasi sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10



Dalam dokumen ada data yang tertulis dengan urutan sebagai berikut:

12, 15, 7, 10, 5, 2, 17, 25, 9, 20, 35, 28, 14, 11, 999 (999 sebagai end of data)

Susun algoritma untuk menginput data dalam dokumen diatas, satu persatu (mulai dari urutan paling kiri) dan menyimpannya kedalam array sehingga isi array menjadi:

a. Yang disimpan hanya bila nilai yang diinput yang bernilai **genap**.

0	1	2	3	4	5	6	7	8	9	10
12	10	2	10	28	14					

b. Yang disimpan bila data yang diinput bernilai **lebih besar dari 10**.

0	1	2	3	4	5	6	7	8	9	10
12	15	17	25	20	35	28	14	11		



KESIMPULAN

1. Array adalah sekumpulan variabel yang memiliki tipe data yang sama dan dinyatakan dengan nama yang sama.
2. Array menggunakan indeks integer untuk menentukan urutan elemen-elemennya, dimana elemen pertama dimulai dengan indeks 0, elemen kedua 1, dst
3. Array memiliki ukuran yang tetap dalam arti tidak dapat membesar atau mengecil ukurannya setelah didefinisikan.





FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BUDI LUHUR

Jl. Raya Ciledug, Petukangan Utara, Pesanggrahan

Jakarta Selatan, 12260

Telp: 021-5853753 Fax : 021-5853752

<http://fti.budiluhur.ac.id>