



# Array

(TIB11 – Struktur Data)

Pertemuan 3, 4

# Sub-CPMK

- Mahasiswa mampu menggunakan array untuk menyimpan dan mengakses data (C3, A3)

# Materi

- Array n Dimensi
- Index Array
- Record Dengan Array



1.

Array n Dimensi

# Pengertian Array

- Array atau Larik adalah sejumlah data secara berurutan
- Mempunyai susunan elemen yang sama
- Setiap array dapat diakses menggunakan indeks yang menyatakan urutan penempatan data pada array
- Secara umum, Array adalah sekumpulan item-item data yang homogen yang dapat dipilih menggunakan indeks pada saat program dijalankan
- Array secara sederhana dibentuk dari tipe data primitif, sehingga membentuk sederetan data dengan tipe data yang sama
- Array dapat dibentuk dari struktur /record

# Tipe data array

- Tipe data array adalah jenis data yang mewakili kumpulan elemen (nilai atau variabel), masing-masing dipilih oleh satu atau beberapa indeks (kunci identifikasi) yang dapat dihitung pada run time selama eksekusi program. Koleksi seperti ini biasanya disebut variabel array, nilai array, atau array sederhana.
- Dengan analogi dengan konsep matematis vektor dan matriks, tipe array dengan satu dan dua indeks sering disebut tipe vektor dan tipe matriks.

# Struktur Data Array

- Struktur data array, atau hanya array, adalah struktur data yang terdiri dari kumpulan elemen (nilai atau variabel), masing-masing diidentifikasi oleh setidaknya satu indeks array atau kunci. Sebuah array disimpan sehingga posisi masing-masing elemen dapat dihitung dari tupel indeksnya dengan rumus matematika.
- Jenis struktur data yang paling sederhana adalah array linier, disebut juga array satu dimensi.

# Dimensi Array

- Array dapat tersusun dalam 1 dimensi, 2 dimensi, 3 dimensi bahkan lebih
- Array dengan 1 dimensi biasa digunakan untuk menyatakan himpunan atau sejumlah record
- Array dengan 2 dimensi biasa digunakan untuk menyatakan sekumpulan himpunan matriks atau tabel
- Array dengan 3 dimensi biasa digunakan untuk menyatakan sekumpulan matriks atau tabel



# Array 1Dimensi

Array  
index

Jakarta	Surabaya	Medan	Manado	...	Denpasar
i=0	i=1	i=2	i=3	...	i=n

index

Array

i=0

Jakarta

i=1

Surabaya

i=2

Medan

i=3

Manado

...

...

i=n

Denpasar

# Array 2 Dimensi

	j=0	j=1	j=2	j=3	...	j=n
i=0	Jakarta	Surabaya	Medan	Manado	...	Denpasar
i=1	New York	Manhattan	California	Kentucky	...	Washington
i=2	Tokyo	Osaka	Kyoto	Hiroshima	...	Nagasaki
i=3	Bangkok	Pattaya	Chiangmai	Mukdahan	...	Krbai
...	...	...	...	...	...	...
i=n	Beijing	Shanghai	Guangzhou	Shenzhen	...	Chengdu

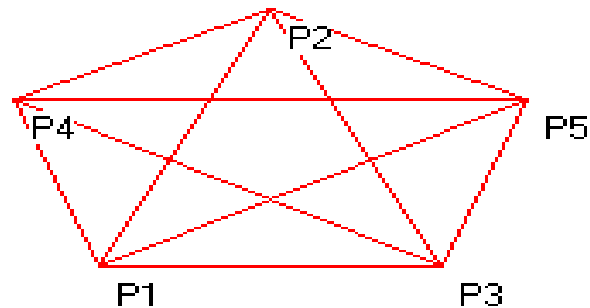
# Implementasi Tabel

- Sebuah Tabel dapat diterapkan dengan struct yang diimplementasikan dengan array 1D
- Tabel dengan struct yang dideklarasikan pada array 1D menjadikan fields sebagai kolom dan index array sebagai baris

# Implementasi Matriks

- Matriks dapat diterapkan dengan menggunakan array 2 dimensi
- Implementasi matriks dengan struct yang dideklarasikan pada array 1 dimensi untuk menyatakan dapat juga dilakukan dengan fields merupakan array juga untuk menyatakan kolom

# Contoh Implementasi Tabel dan Matriks



Input Koordinat

P1	50	50
P2	100	150
P3	150	50
P4	25	115
P5	175	115

Input Arah

	P1	P2	P3	P4	P5
P1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
P2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
P3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
P4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
P5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

# Contoh Implementasi Tabel dan Matriks (cont.)

Pada Contoh ditampilkan sebuah grafik dengan koordinat-koordinat dan arah vektornya.

- P1 dengan koordinat(50,50)
- P2 dengan koordinat(100,150)
- P3 dengan koordinat(150,50)
- P4 dengan koordinat(25,115)
- P5 dengan koordinat(175,115)

# Contoh Implementasi Tabel dan Matriks (cont.)

- Jika digambarkan sebagai tabel maka tabel vektor adalah sbb

Input Koordinat

<b>P1</b>	50	50
<b>P2</b>	100	150
<b>P3</b>	150	50
<b>P4</b>	25	115
<b>P5</b>	175	115



Tabel Vektor

Vektor	X	Y
P1	50	50
P2	100	150
P3	150	50
P4	25	115
P5	175	115

# Contoh Implementasi Tabel dan Matriks (cont.)

Dengan menggunakan Record yang dideklarasikan dengan array, maka tabel vektor dapat digambarkan dengan cara

- Label Vektor dapat diwakili dengan Indeks array
- Kolom X diwakili dengan fields X pada tiap record
- Kolom Y diwakili dengan fields Y pada tiap record

## Implementasi Dengan Record dan Array

Indeks	X	Y
1	50	50
2	100	150
3	150	50
4	25	115
5	175	115



# Contoh Implementasi Tabel dan Matriks (cont.)

Dengan menggunakan array 2 Dimensi maka tabel vektor dapat digambarkan dengan cara:

- Label Vektor dapat diwakili dengan Indeks array dimensi pertama
- Kolom X diwakili array dimensi ke 2 indeks ke 1
- Kolom Y diwakili array dimensi ke 2 indeks ke 2

Implementasi dengan  
Arrad 2D

Indeks	1	2
1	50	50
2	100	150
3	150	50
4	25	115
5	175	115

↑  
diasumsikan sebagai  
kolom Field X

↑  
diasumsikan sebagai  
kolom Field Y

# Contoh Implementasi Tabel dan Matriks (cont.)

- Arah vektor diimplementasikan dengan matriks arah (ini akan dibahas pada materi 8 mengenai Graph), dimana baris merupakan asal vektor dan kolom merupakan tujuan.

	P1	P2	P3	P4	P5
P1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
P2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
P3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
P4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
P5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



	P1	P2	P3	P4	P5
P1	0	1	0	1	0
P2	0	0	1	0	1
P3	1	0	0	1	0
P4	0	1	0	0	1
P5	1	0	1	0	0

# Contoh Implementasi Tabel dan Matriks (cont.)

Dengan menggunakan array 2 Dimensi maka matriks arah dapat digambarkan dengan cara:

- Array dimensi pertama digunakan sebagai indeks baris
- array dimensi kedua sebagai indeks kolom
- Dapat juga diterapkan dengan cara sebaliknya.

## Implementasi dengan array 2 dimensi

array dimensi ke dua

j

		1	2	3	4	5
1	0	1	0	1	0	
2	0	0	1	0	1	
3	1	0	0	1	0	
4	0	1	0	0	1	
5	1	0	1	0	0	

i

array dimensi pertama

# Contoh Implementasi Tabel dan Matriks (cont.)

Dengan menggunakan record yang berisi field array yang dideklarasikan dengan array 1 Dimensi maka matriks arah dapat digambarkan dengan cara:

- Array sebagai baris
- Field berupa array sebagai kolom pada tiap barisnya

Struct berisi field array yang dideklarasikan sebagai array 1 dimensi

		Field Array				
		1	2	3	4	5
Array i	1	0	1	0	1	0
	2	0	0	1	0	1
	3	1	0	0	1	0
	4	0	1	0	0	1
	5	1	0	1	0	0

# Contoh Array 2 D dalam C

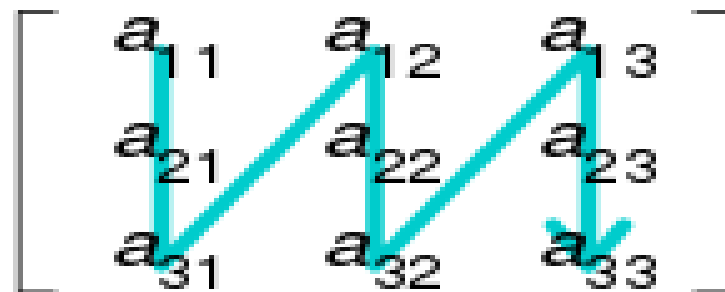
- `#include <stdio.h>`
- 
- `int main(void)`
- `{`
- `int matrix[2][3] = {{1,2,3},{7,8,9}};`
- 
- `printf("Isi array matrix: \n");`
- `printf("%d %d %d \n", matrix[0][0], matrix[0][1], matrix[0][2]);`
- `printf("%d %d %d \n", matrix[1][0], matrix[1][1], matrix[1][2]);`
- 
- `return 0;`
- `}`

# Row- and column-major order

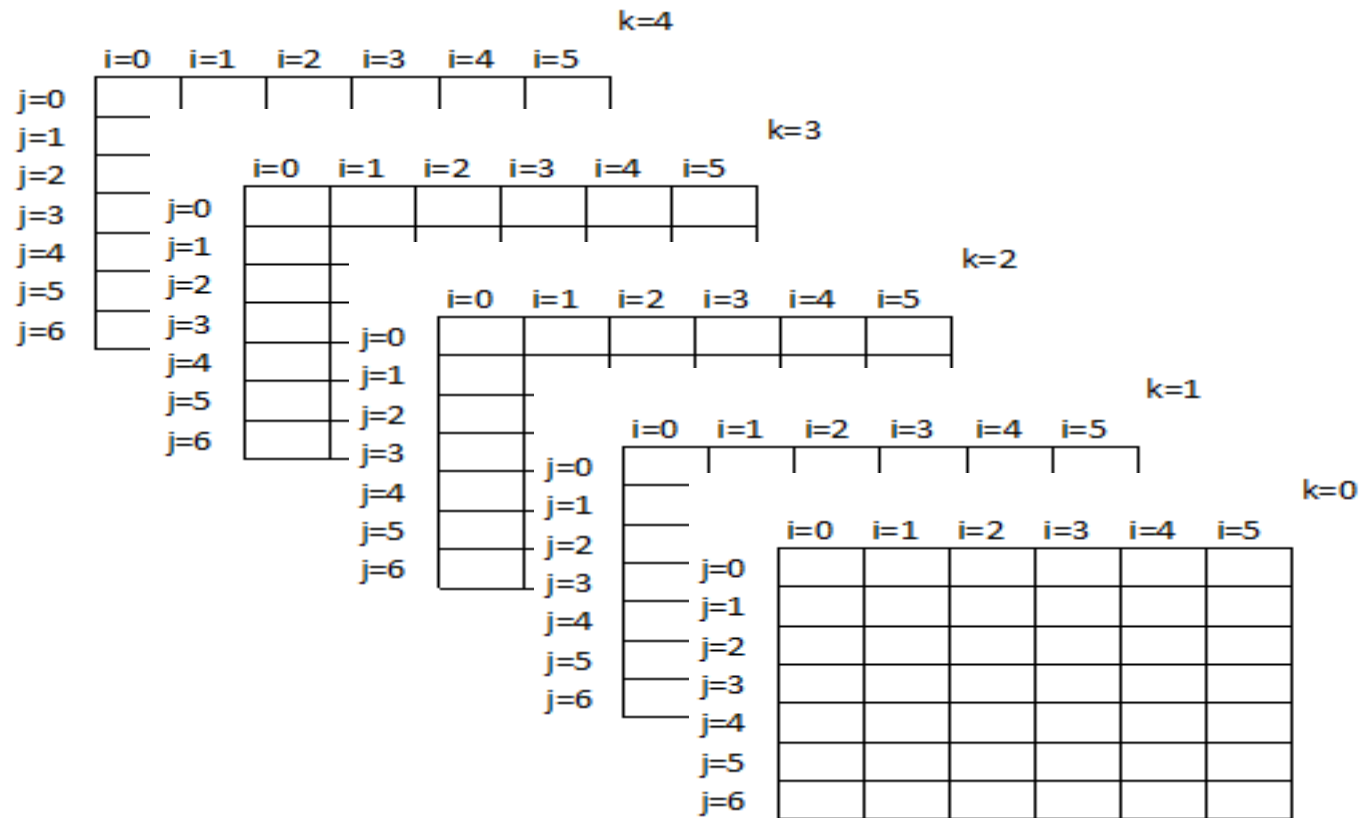
Row-major order



Column-major order



# Array 3 Dimensi



# Array 1D (pascal)

## Declaration

**namaArray** : **array** [awalIndex..akhirIndex] of tipevariabel;

Var

table1d : array[1..10] of integer;

**Assignment**      → table1d[8] := 1000;

//mengisi array ke 8 dengan 1000

**Accessing**        → temp := table1d[8];

//mengambil array ke 8 dan menyimpan pada variabel temp



# Array 2D (pascal)

## Declaration

```
namaArray : array [awalIndexDimensi1..akhirIndexDimensi1,  
awalIndexDimensi2..akhirIndexDimensi2] of tipevariabel;
```

//contoh array 2D berukuran 5 x 10

Var

```
table2d : array[1..5, 1..10] of byte;
```

**Assignment**      → table2d[1,3] := 88;

//mengisi array baris 1 kolom 3 dengan 88

**Accessing**      → temp := table2d[1,3];

//mengambil array baris 1 kolom 3 dan menyimpan pada variabel temp

# Array 3D (pascal)

## Declaration

```
namaArray : array [awalIndexDimensi1..akhirIndexDimensi1,  
awalIndexDimensi2..akhirIndexDimensi2, awalIndexDimensi3..akhirIndexDimensi3] of  
tipevariabel;
```

//contoh array 3D berukuran 5 x 4 x 6

Var

```
table3d : array [1..5, 1..4, 1..6] of byte;
```

**Assignment**      → table3d[1,4,3] := 100;

//mengisi array baris 1 kolom 4 tabel 3 dengan 100

**Accessing**      → temp := table3d[1,4,3];

//mengambil array baris 1 kolom 4 tabel 3 dan menyimpan pada variabel temp

# Array 1D (C)

## Declaration

```
tipeVariabel namaArrayname[ukuranArray] ;
```

```
//deklarasi array berukuran 10 dengan indeks 0..9
```

```
//Catatan: pada C index array dimulai dari 0
```

```
int table1d[10];
```

## Assignment

```
table1d[7] = 1000; //mengisi array ke 7 dengan 1000
```

## Accessing

```
temp = table1d[7]; //mengambil data dari array ke 7 dan menyimpan pada variabel temp
```

# Array 2D (C)

## Declaration

```
typeVar namaArrayname[ukuranArraydimensi1][ukuranArraydimensi2];  
//deklarasi array berukuran 10 x 20 bertipe data int  
int table2d[10][20];
```

## Assignment

```
//mengisi array baris 1 kolom 4 dengan 88  
table3d[1][4] = 88;
```

## Accessing

```
//mengambil data array pada baris 1 kolom 4 dan menyimpan pada variabel temp  
temp = table2d[1][4];
```

# Array 3D (C)

## Declaration

```
typeVar namaArray[ukuranArrayDim1][ukuranArrayDim2][ukuranArrayDim3];  
//deklarasi array berukuran 10 x 20 x 5 bertipe int  
    int table3d[10][20][5];
```

## Assignment

```
//mengisi baris 1 kolom 4 tabel 5 dengan 100  
    table3d[1][4][5] = 100;
```

## Accessing

```
//mengambil data array pada baris 1 kolom 4 tabel 5  
//dan menyimpan pada variabel temp  
    temp = table3d[1][4][5]; //Accessing
```



2.

Index Array



3.

Record Dengan Array

# Record with array - PASCAL

- Record definition

**Type**

```
RecordName = Record  
  Var1Name : vartype;  
  Var2Name : vartype;  
  VarnName : vartype;  
End;
```

- Array declaration

```
DataCell : array[1..250] of RecordName;
```

- Assignment

```
DataCell[ArrayNum].VarName := value;
```

- Accessing

```
DataCell[ArrayNum].VarName
```

- Example

```
//record definition
```

**Type**

```
TheCell=Record
```

```
  Name : string;  
  Age : Integer;
```

```
End;
```

```
//Declaration
```

```
Var
```

```
  DataMhs : array[1..250] of TheCell;
```

```
Begin
```

```
//Assignment
```

```
DataMhs[1].Name := "Doraemon";
```

```
DataMhs[1].Age := 19;
```

```
//Accessing
```

```
writeln(DataMhs[1].Name);
```

```
writeln(DataMhs[1].Age);
```

```
End.
```



# Record with array - C

- Record definition

```
struct StructName
{
    vartype Var1Name;
    vartype Var2Name;
    vartype VarNName;
};
```

- Array declaration

```
struct StructName DataCell[ArraySize];
```

- Assignment

```
DataCell[ArrayNum].VarName = value;
```

- Accessing

```
DataCell[ArrayNum].VarName
```

- Example

```
//Struct definition
struct TheCell
{
    char Name[10];
    int Age;
};
//Declaration
struct TheCell DataMhs[250];

void main()
{
    //Assignment
    strcpy(DataMhs[1].Name, "Doraemon");
    DataMhs[1].Age = 19;
    //Accessing
    printf("%s", DataMhs[1].Name);
    printf("%d", DataMhs[1].Age);
}
```

# Ringkasan

- Array atau Larik adalah sejumlah data secara berurutan yang mempunyai susunan elemen yang sama
- Array dapat diakses menggunakan indeks yang menyatakan urutan penempatan data pada array
- Array dapat dibentuk dari struktur / record
- Array dapat tersusun dalam 1 dimensi, 2 dimensi, 3 dimensi bahkan lebih
- Array dengan 1 dimensi biasa digunakan untuk menyatakan himpunan atau sejumlah record
- Array dengan 2 dimensi biasa digunakan untuk menyatakan sekumpulan himpunan matriks atau tabel
- Array dengan 3 dimensi biasa digunakan untuk menyatakan sekumpulan matriks atau tabel



*Terimakasih*

***TUHAN Memberkati Anda***

Teady Matius Surya Mulyana (tmulyana@bundamulia.ac.id)