

Array / Larik / Variabel Berindek

Array 2 Dimensi

- Pada Array 1 Dimensi :

1	11	32	17	25	12	66
---	----	----	----	----	----	----

- Terdiri dari 1 baris, berisi beberapa data, semuanya memiliki tipe data yang sama

Array 2 Dimensi

- Pada Array 2 Dimensi :

12	17	22	14
10	5	13	5

- Terdiri lebih dari 1 baris dan 1 kolom, berisi beberapa data yang semuanya memiliki tipe data yang sama

Array 2 Dimensi

- Terdiri dari baris dan kolom

	0	1	2	3	→ Kolom, 4
0	12	17	22	14	Dimensi Array dinyatakan dalam Baris x Kolom
1	10	5	13	5	
↓ Baris, 2					<u>Array 2 x 4</u>

Array 2 Dimensi

12	17
10	5

12
10
17

12	17
10	5
17	11

1	11	32
---	----	----

Deklarasi Array 2 Dimensi

Tipe-data **nama-array**[jumlah baris][jumlah kolom]

tipe-data : tipe data dari elemen array

nama-array : nama dari variabel array

jumlah baris : jumlah baris elemen array

jumlah kolom : jumlah kolom elemen array

Deklarasi Array 2 Dimensi

- Contoh :

int arrayku[2][3];

→ Mendeklarasikan array berukuran 2x3,
bertipe integer

	0	1	2
0			
1			

Deklarasi Array 2 Dimensi

- Contoh :

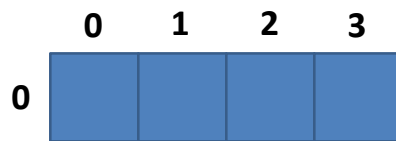
char peserta[10][50];

→ Mendeklarasikan array berukuran 10x50
bertipe char

Deklarasi Array 2 Dimensi

- Contoh :

int data[1][4]; → bagaimanakah array yang terbentuk ?



Sama saja dengan → **int data[4];**

Inisialisasi Array 2 Dimensi

- Inisialisasi bisa dilakukan saat variabel dideklarasikan
- Untuk Array 1 Dimensi, pemberian nilai dengan tanda **{ }**
- Dengan Array 2 Dimensi sama saja, hanya ada tambahan tanda **{ }** untuk masing-masing barisnya

Inisialisasi Array 2 Dimensi

- Array 1 Dimensi :

int data[3] = {30, 40, 50};

30	40	50
----	----	----


- Array 2 Dimensi :

int data[2][3] = { {10,20,30}, {40,50,60} };

10	20	30
40	50	60

Inisialisasi Array 2 Dimensi

int data[2][3] = { {10,20,30}, {40,50,60} };


 Baris ke 0 Baris ke 1

	0	1	2
0	10	20	30
1	40	50	60

Inisialisasi Array 2 Dimensi

- Jumlah baris dan kolom bisa tidak dicantumkan asalkan array langsung diinisialisasikan

```
int data[][] = { {10,20,30}, {40,50,60} };
```

→ Array berukuran 2x3, bertipe integer

Inisialisasi Array 2 Dimensi

```
int data[2][3] = {{10, 20, 30}, {40, 50, 60}};
```

- Untuk mempermudah penulisan dan pembacaan, inisialisasi dapat dilakukan dengan penulisan berikut :

```
int data[2][3] = {{10, 20, 30},  
                  {40, 50, 60}};
```

Inisialisasi Array 2 Dimensi

- Khusus untuk array 2 dimensi bertipe char, inisialisasi dapat dilakukan dengan cara-cara berikut :

```
char nama[2][6] = {{ 'm', 'a', 'r', 'k',  
                    'k', 'e', 'v', 'i', 'n' }};
```

```
char nama[2][6] = { "mark",  
                    "kevin" };
```

Pengaksesan Array 2 Dimensi

- Elemen dalam array 2 dimensi diakses dengan penanda baris dan kolom

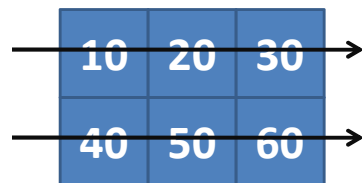
- Contoh :

	0	1	2
0	10	20	30
1	40	50	60

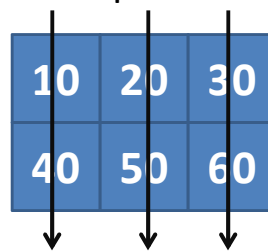
→ diakses dengan : `data[0][1];`

Pengaksesan Array 2 Dimensi

- Urutan pengaksesan tidak harus baris-per-baris, tapi bisa kolom-per-kolom sesuai kebutuhan
- Baris-per-baris :



Kolom-per-kolom



Operasi Pada Array 2 Dimensi

- Array 2 dimensi sering disebut matriks
- Karena itu, operasi pada array 2 dimensi pada umumnya adalah operasi matriks, seperti menjumlahkan, mengurangi, dan mengkalikan dua matriks, inverse matriks, transpose matriks dan sebagainya

Operasi Pada Array 2 Dimensi

- Untuk menyalin isi matriks ke matriks lainnya harus menyalin setiap elemen (baris dan kolom)

```
int data[2][3] = {{1,2,3}, {2,2,2}};
int salinan[2][3];
salinan = data;      → Proses ini salah !!
```

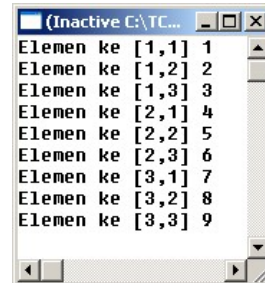
Operasi Pada Array 2 Dimensi

```
...
int data[2][3] = {{1,2,3}, {2,2,2}};
int salinan[2][3];
for(int b=0; b<2; b++) {
    for(int k=0; k<3; k++) {
        salinan[b][k] = data[b][k];
    }
}
...
```

Operasi Matriks : Input

```
#include <stdio.h>

void main() {
    int matriks[3][3];
    for(int b=0; b<3; b++) {
        for(int k=0; k<3; k++) {
            printf("Elemen ke [%d,%d] ", b+1, k+1);
            scanf("%d", &matriks[b][k]);
        }
    }
}
```



Operasi Matriks : Cetak

```
#include <stdio.h>

void main() {
    int matriks[3][3] = {{1,2,3},
                        {4,5,6},
                        {7,8,9}};
    for(int b=0; b<3; b++) {
        for(int k=0; k<3; k++) {
            printf("%d ", matriks[b][k]);
        }
        printf("\n");
    }
}
```



Operasi Matriks : Transpose

- Operasi ini membalik posisi elemen-elemen dalam matriks, baris menjadi kolom dan sebaliknya, kolom menjadi baris

1	2	3
4	5	6
7	8	9

→

1	4	7
2	5	8
3	6	9

Operasi Matriks : Transpose

```
#include <stdio.h>

void main() {
    int matriks[3][3] = {{1,2,3},
                        {4,5,6},
                        {7,8,9}};

    int transpose[3][3];
    for(int b=0; b<3; b++) {
        for(int k=0; k<3; k++) {
            transpose[k][b] = matriks[b][k];
        }
    }
}
```

Operasi Matriks : Tambah & Kurang

- Untuk menjumlahkan atau mengurangi dua matriks, syaratnya adalah :
 - Dimensi kedua array sama (ukuran baris dan kolomnya sama)

Operasi Matriks : Tambah & Kurang

1	2	3		1	4	7		2	6	10
4	5	6	+	2	5	8	=	6	10	14
7	8	9		3	6	9		10	14	18