

# Pertemuan 3

## **ARRAY DIMENSI BANYAK**

### 3. ARRAY DIMENSI TIGA (Three Dimensional Array)

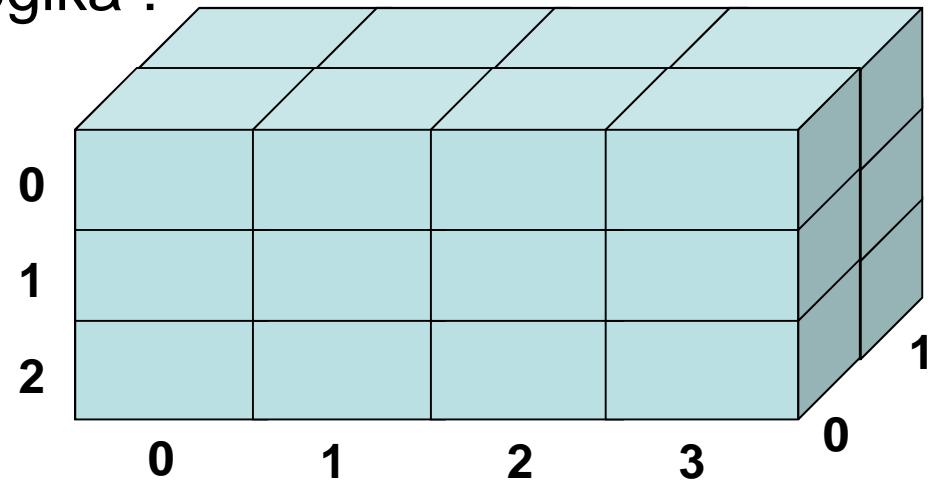
Digunakan untuk mengelola data dalam bentuk 3 dimensi atau tiga sisi.

Deklarasi :

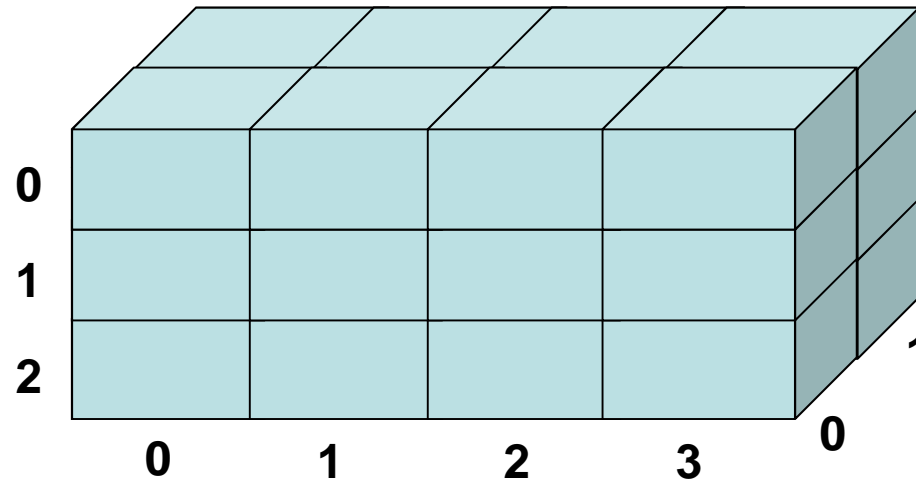
Type\_Data Nama\_Variabel [index1] [index2] [index3];

Misal : int A [3][4][2];

Penggambaran secara Logika :



- Contoh aplikasi untuk Array dimensi 3 adalah seperti program input data mahasiswa per kelas dan per jurusan



Menentukan jumlah elemen dalam Array dimensi 3 :

$$\prod_{i=1}^n \pi(\text{index array})$$

$\pi$  = Perkalian dari statemen sebelumnya

Contoh :

Suatu Array X dideklarasikan sbb :

int A [3][4][2]; maka jumlah elemen Array dimensi tiga tersebut adalah :

$$(3) * (4) * (2) = 24$$

## PEMETAAN (MAPPING) ARRAY DIMENSI TIGA KE STORAGE

Rumus :

$$\text{@M}[n][m][p] = \text{M}[0][0][0] + \{((n-1) * (\text{index1})) + ((m-1) * (\text{index2})) + ((p-1) * (\text{index3}))\} * L$$

Contoh :

Suatu Array A dideklarasikan sebagai berikut :

int A [2][4][3], dengan alamat awal index A[0][0][0] berada di  $0011_{(H)}$  dan ukuran type data int = 2 Tentukan berapa alamat array di A[2][3][2] ?

## Contoh Pemetaan :

### Penyelesaian :

1. Tentukan jumlah elemen array A [2][4][3]

$$= (2) * (4) * (3)$$

$$= 24$$

2.  $@M[n][m][p] = M[0][0][0] + \{((n-1) * (index1)) + ((m-1) * (index2)) + ((p-1) * (index3))\} * L$

$$\begin{aligned} A[2][3][2] &= 0011_{(H)} + \{((2-1) * 4 * 3) + ((3-1) * 3) + (2-1)\} * 2 \\ &= 0011_{(H)} + \{12 + 6 + 1\} * 2 \\ &= 0011_{(H)} + 38_{(D)} \longrightarrow 26_{(H)} \\ &= 0011_{(H)} + 26_{(H)} \\ &= 0037_{(H)} \end{aligned}$$

# TRINGULAR ARRAY (ARRAY SEGITIGA)

Tringular Array dapat merupakan Upper Tringular (seluruh elemen di bawah diagonal utama = 0), ataupun Lower Tringular (seluruh elemen di atas diagonal utama = 0).

Dalam Array Lower Tringular dengan N baris, jumlah maksimum elemen  $\neq 0$  pada baris ke-l adalah = l, karenanya total elemen  $\neq 0$ , tidak lebih dari

$$\sum_{l=1}^N l = N(N+1) / 2$$

$$\begin{array}{c}
 \left[ \begin{array}{cccccc}
 X & X & X & X & X & X \\
 0 & X & X & X & X & X \\
 0 & 0 & X & X & X & X \\
 0 & 0 & 0 & X & X & X \\
 0 & 0 & 0 & 0 & X & X \\
 0 & 0 & 0 & 0 & 0 & X
 \end{array} \right] \\
 \text{(a)}
 \end{array}
 \qquad
 \begin{array}{c}
 \left[ \begin{array}{cccccc}
 X & 0 & 0 & 0 & 0 & 0 \\
 X & X & 0 & 0 & 0 & 0 \\
 X & X & X & 0 & 0 & 0 \\
 X & X & X & X & 0 & 0 \\
 X & X & X & X & X & 0 \\
 X & X & X & X & X & X
 \end{array} \right] \\
 \text{(b)}
 \end{array}$$

Gambar (a) Upper Triangular Array  
 (b) Lower Triangular Array



Diketahui suatu array segitiga atas memiliki 3 baris dan kolom, tentukan berapakah jumlah elemen yang bukan nol pada array tersebut.

$$\begin{aligned} I &= N(N+1) / 2 \\ I &= 3(3+1) / 2 \\ &= 12 / 2 \\ &= 6 \end{aligned}$$

Contoh bentuk array nya adalah seperti dibawah ini :

10	20	30		5	10	15	Dan lain-lain
0	40	50		0	20	25	
0	0	60		0	0	30	

Suatu Array Upper Tringular dan Array Lower Tringular dapat dengan order yang sama, dapat disimpan sebagai suatu array dengan order yang berbeda, Contohnya :

$$A = \begin{vmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{vmatrix} \quad B = \begin{vmatrix} 7 & 0 & 0 \\ 8 & 9 & 0 \\ 11 & 12 & 13 \end{vmatrix}$$

dapat disimpan sebagai Array C berorder (3 X 4)

$$\begin{vmatrix} 7 & 1 & 2 & 3 \\ 8 & 9 & 4 & 5 \\ 11 & 12 & 13 & 6 \end{vmatrix}$$

# SPARSE ARRAY (ARRAY JARANG)

Suatu Array yang sangat banyak elemen nol-nya, contohnya adalah Array A pada Gambar berikut :

0	0	0	0	1	0	0	2
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	4	0	0	0	0
0	0	0	0	0	0	0	2