


Array



DEFINISI

- Tipe data terstruktur yang merujuk kepada sebuah atau sekumpulan elemen yang mempunyai tipe data yang sama melalui indeks.
- Tabel, vektor atau larik 
- Elemen dari array dapat diakses langsung jika dan hanya jika indeks terdefinisi (telah ditentukan nilainya sesuai dengan domain yang didefinisikan untuk indeks tersebut)

CONTOH

- $A[1], A[2], A[3], A[4], A[5], A[6], A[7]$

Indeks Larik / Array

Nama Larik / Array

A[1]

ARRAY

1
Dimensi

2
Dimensi

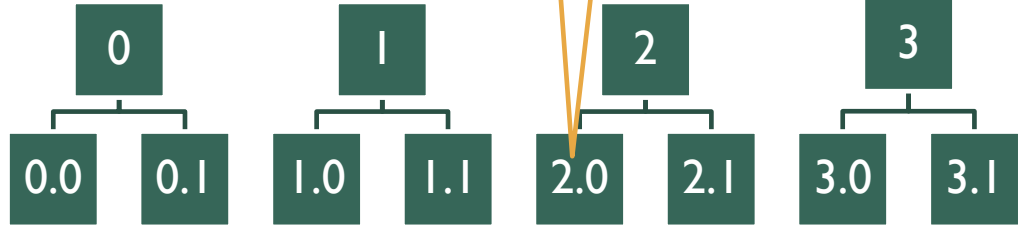
Multi
Dimensi

ARRAY

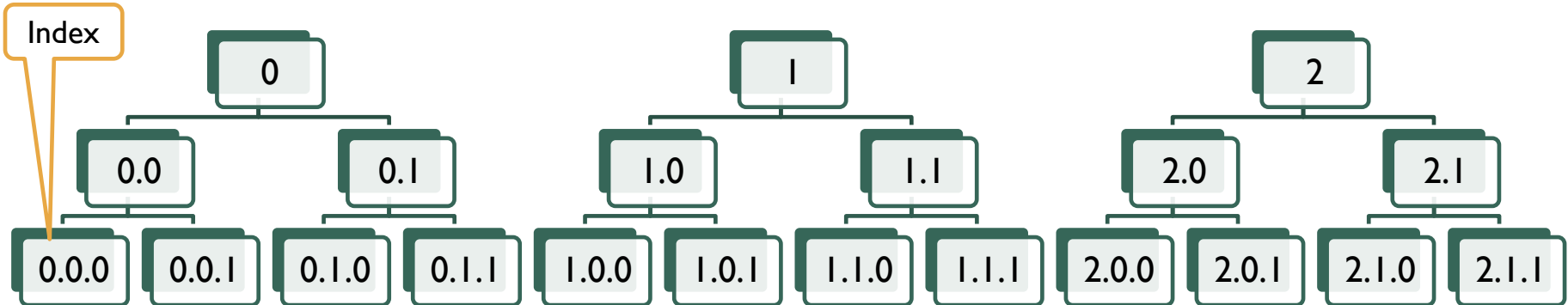
Array 1 Dimensi = [4]



Array 2 Dimensi = [4] [2]



Array 3 Dimensi = [2] [2] [2]



ARRAY 2 DIMENSI

Perbedaan dengan array 1 dimensi terletak pada indeks yang dimilikinya

ARRAY 2 DIMENSI – DEKLARASI ARRAY

Algoritma	Java
namaarray : array [1...jumlahindeks_pertama] [1...jumlahindeks_kedua] of typedata	typedata namaarray [] [] = new typedata [jumlahindeks_pertama] [jumlahindeks_kedua];
Algoritma	Java
Matriks : array [1...2][1...3] of integer	Int Matriks [][] = new int [2][3];

ARRAY 2 DIMENSI – MENGISI INDEKS ARRAY

Algoritma	Java
<code>namaarray [indekspertama][indekskedua] ← nilai</code>	<code>namaarray [indekspertama][indekskedua] = nilai;</code>

Algoritma	Java
<code>Matriks[1][3] ← 2</code>	<code>Matriks[0][2] =2;</code>

ARRAY 2 DIMENSI – MENGAkses NILAI PADA INDEKS TERTENTU

Algoritma	Java
<code>namaarray [indekspertama] [indekskedua]</code>	<code>namaarray [indekspertama] [indekskedua];</code>

Algoritma	Java
<code>Matriks[2] [2]</code>	<code>Matriks[2] [2];</code>

CONTOH

```
1  #include <stdio.h>
2
3  int main(void) {
4      // Deklarasi dan inisialisasi array dua dimensi
5      int bilangan[2][2];
6      bilangan[0][0] = 100;
7      bilangan[0][1] = 101;
8      bilangan[1][0] = 110;
9      bilangan[1][1] = 111;
10
11     // Menampilkan isi array bilangan
12     printf("Isi array bilangan: \n");
13     printf("%d, %d \n", bilangan[0][0], bilangan[0][1]);
14     printf("%d, %d \n", bilangan[1][0], bilangan[1][1]);
15
16     return 0;
17 }
```

CONTOH ARRAY DUA DIMENSI (ALGORITMA)

```
1.  ALGORITMA
2.  /* Menyiapkan dan memasukkan nilai dalam array dua dimensi
3.  I.S : array dalam keadaan kosong
4.  F.S : menampilkan nilai yang disimpan dalam array dengan menggunakan struktur
5.  pengulangan */
6.  KAMUS DATA
7.  A : array[0..2,0..4] of integer
8.  i,j,k : integer
9.  BEGIN
10. k := 0;
11. /* Memasukkan nilai dalam array */
12. for i := 0 to 3 do
13.   for j := 0 to 4 do
14.    A[i, j] := k + 2;
15.    k := k + 2;
16.   endfor;
17. endfor;
18. /* Menampilkan kembali nilai pada array */
19. for i := 0 to 3 do
20.   for j := 0 to 4 do
21.    output('A[' , i , ',' , j , ']' = ' , A[i, j]);
22.   endfor;
23. endfor;
24. END.
```

CONTOH ARRAY DUA DIMENSI (ALGORITMA)

```
1- /* Menyiapkan dan memasukkan nilai dalam array dua dimensi
2   I.S : array dalam keadaan kosong
3   F.S : menampilkan nilai yang disimpan dalam array dengan menggunakan struktur pengulangan */
4 #include <stdio.h>
5 int main() {
6     int A[3][5]; /* deklarasi array dua dimensi */
7     int i, j, k;
8     k = 0;
9     /* memasukkan data dalam array dua dimensi */
10    for (i = 0; i <= 2; i++) {
11        for (j = 0; j <= 4; j++) {
12            A[i][j] = k + 2;
13            k += 2;
14        } /* endfor loop j */
15    } /* endfor loop i */
16    /* menampilkan kembali nilai array dua dimensi */
17    for (i = 0; i <= 2; i++) {
18        for (j = 0; j <= 4; j++) {
19            printf("A[%i,%i] = %i\n", i, j, A[i][j]);
20        } /* endfor loop j */
21    } /* endfor loop i */
22    printf("Tekan Enter untuk melanjutkan...");
23    getchar(); /* menahan tampilan pada layar dengan menunggu input Enter */
24    return 0;
25 }
```

Output

/tmp/IbuJqYvxvP.o

A[0,0] = 2
A[0,1] = 4
A[0,2] = 6
A[0,3] = 8
A[0,4] = 10
A[1,0] = 12
A[1,1] = 14
A[1,2] = 16
A[1,3] = 18
A[1,4] = 20
A[2,0] = 22
A[2,1] = 24
A[2,2] = 26
A[2,3] = 28
A[2,4] = 30

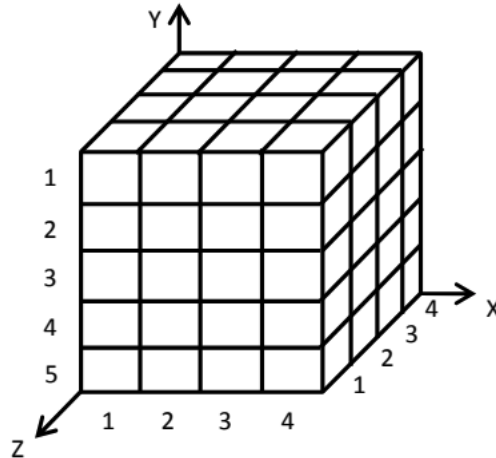
Tekan Enter untuk melanjutkan...

=== Session Ended. Please Run the code again ===

ARRAY MULTI-DIMENSI

- Dalam menggambarkan array multidimensi, hanya terbatas hingga dimensi ke-3, yakni dengan menggunakan bangun ruang, namun dalam kenyataannya, tipe data array ini dapat dibentuk menjadi lebih dari tiga dimensi atau menjadi n-dimensi.

Representasi array 3 (tiga) dimensi



LATIHAN

- I. Sebuah perusahaan ayam goreng dengan nama “SAKTI FRIED CHICKEN” yang telah lumayan banyak pelanggannya, ingin dibantu dibuatkan program untuk membantu kelancaran usahanya. “SAKTI FRIED CHICKEN” mempunyai daftar harga ayam sebagai berikut :

Kode Jenis Harga

D Dada Rp. 12500

P Paha Rp. 10000

S Sayap Rp. 8500

Buatlah programnya dengan ketentuan:

- Setiap pembeli dikenakan pajak sebesar 10% dari pembayaran.
- Banyak Jenis, Jenis Potong dan Banyak Beli diinput.
- Tampilan yang diinginkan sebagai berikut:

Layar Masukkan
SAKTI FRIED CHICKEN

Kode Jenis Harga

D Dada Rp. 12500

P Paha Rp. 10000

S Sayap Rp. 8500

Banyak Jenis : ... <diinput>

Jenis Ke - ... <proses counter>

Jenis Potong [D/P/S] : ... <diinput>

Banyak Potong : ... <diinput>

<<Terus berulang tergantung Banyak Jenis>>

Layar Keluaran

SAKTI FIRED CHICHEN

No. Jenis Harga Bayak. Jumlah
 Potong Satuan Beli Harga

... Rp

... Rp

Jumlah Bayar Rp

Pajak 10% Rp

Total Bayar Rp