

Algoritma dan Pemrograman Lanjut

Pertemuan Ke-2 Array (Larik)



Disusun Oleh :
Wilis Kaswidjanti, S.Si., M.Kom.

**Jurusan Teknik Informatika
Fakultas Teknologi Industri
Universitas Pembangunan Nasional “Veteran”
Yogyakarta**

Algoritma dan Pemrograman Lanjut

Judul Materi : Array (Larik)

Deskripsi Materi : Materi ini mencakup subpokok bahasan array dimensi satu, dimensi dua, perhitungan matriks, dan pengenalan dimensi n dengan menggunakan tipe data array

Tujuan Instruksional Khusus :

1. Mendefinisikan dan membuat matriks dengan menggunakan tipe data array
2. Mengimplementasikan tipe data array untuk matrik
3. Menggunakan definisi matriks untuk perhitungan matriks

Referensi :

- Buku Teks
Munir, Rinaldi (2005), *Algoritma dan Pemrograman dalam Bahasa Pascal dan C*, Buku 2, Edisi Ketiga, Penerbit Informatika Bandung, Bab 3, hal 77-113.
Charibaldi, N. (2004), *Modul Kuliah Algoritma Pemrograman II*, Edisi Kedua, Yogyakarta
- Buku Acuan/Referensi
Brassard, Gilles (1999), *Fundamentals of algorithma*, PrinteceHall.
Jarne, Stroustrup B. (1997), *C++ Programming language*, AT &T.
Kristanto, Andri (2003), *Algoritma pemrograman C++*, Graha Ilmu.
Schildt,Herbert (2000), *The Complete Reference C++*, McGraw-Hill.
Sedgewick, R. (2000), *Algoritma Third edition In C part 5*, Addison Wesley.

ARRAY (LARIK)

PENDAHULUAN

Array atau biasa disebut larik adalah koleksi data dimana setiap elemen memakai nama yang sama dan bertipe sama dan setiap elemen diakses dengan membedakan indeks arraynya. Setiap rinci data disebut dengan elemen array. Setiap array dapat dioperasikan menggunakan pemilih elemen yang disebut indeks. Karena itu indeks harus bertipe yang mempunyai keterurutan (ada suksesor dan ada predesor) misal Integre, Boolean, Char.

Jadi array dipakai untuk menyajikan sekumpulan data yang bertipe sama, dan disimpan dengan urutan yang sesuai dengan indeks secara kontinu.

Contoh Array : Untai karakter, daftar nomor telepon, daftar belanja, tabel pajak, vektor, matriks.

ISI

A. Deklarasi Array

Setiap identifier yang akan digunakan harus dipesan dahulu dalam kamus/deklarasi beserta tipe data yang akan ditampungnya. Demikian pula dengan tipe data array beserta tipe elemen-elemen yang akan ditampungnya harus pula dipesan dalam kamus/deklarasi.

Jika kita memesan variabel biasa (tipe data sederhana dan string), maka kita hanya menentukan tipenya, namun jika memesan array, maka harus pula ditentukan banyaknya elemen (misal dari 1..100).

Variabel array dideklarasikan dengan mencantumkan tipe dan nama variable yang diikuti dengan banyaknya lokasi memori yang ingin dibuat.

Contoh :

Dalam Algorima :

c : array[1..5] of integer

Terbentuk 5 variabel c dengan indeks dari 1 smapai dengan 5 bertipe integer.

Dalam bahasa C++ :

```
int c[5];
```

C++ secara otomatis akan menyediakan lokasi memori sesuai dengan yang dideklarasikan, dimana nomor indeks selalu dimulai dari 0.

```
int c[5] = {-12, 0, 20, 85, 1551};
```

Nilai suatu variabel array dapat juga diinisialisasi secara langsung seperti yang terdapat di dalam tanda kurung kurawal pada saat deklarasi di atas.

```
int x[5] = [0];
```

Deklarasi variable array sekaligus mengisi setiap lokasi memorinya dengan nilai 0.

Tipe data yang akan ditampung array bisa sembarang tipe data misal Integer, Boolean, Char, Real, String bahkan yang terstruktur seperti Set, Record atau array lagi.

B. Array Dimensi Satu

Bentuknya (Pseudocode) :

```
nama_var : array [rangeindeks] of tipe
```

Bentuknya (C++):

```
tipe nama_var[ukuran];
```

Dengan :

- tipe : menyatakan jenis elemen array (int, char dan lain-lain)
- rangeindeks : menyatakan indeks awal sampai dengan indeks akhir dari var array yang dibuat
- ukuran : menyatakan jumlah maksimal elemen array

Nilai indeks awal dan indeks akhir yang dideklarasikan di algoritma tidak diperlukan pada deklarasi bahasa C++. Pada bahasa C++ hanya banyak elemennya saja (ukuran). Jadi ukuran = indeksakhir – indeksawal + 1.

Contoh :
Algoritma :

Algoritma PangkatDua

Deklarasi

square : array [1..100] of integer
i,k : integer

Deskripsi

for i \leftarrow 1 to 10 do
 k \leftarrow i + 1
 square[i] \leftarrow k * k
 output("Pangkat dari ",k," adalah ", square[i])
endfor

Program C++ :

```
#include<iostream.h>
main()
{
    int square[100];
    int i,k;

    for(i=0; i<10; i++)
    {
        k = i + 1;
        square[i] = k * k;
        cout<<"\nPangkat dari "<<k<<" adalah "<< square[i];
    }
}
```

Output :

```
Pangkat dari 1 adalah 1
Pangkat dari 2 adalah 4
Pangkat dari 3 adalah 9
Pangkat dari 4 adalah 16
Pangkat dari 5 adalah 25
Pangkat dari 6 adalah 36
Pangkat dari 7 adalah 49
Pangkat dari 8 adalah 64
Pangkat dari 9 adalah 81
Pangkat dari 10 adalah 100
```

C. Array Dimensi Dua

Array dua dimensi merupakan array yang terdiri dari m buah baris dan n buah kolom. Bentuknya dapat berupa matriks atau tabel.

Pada prinsipnya matriks hampir sama dengan array. Matriks adalah sekumpulan informasi bertipe sama yang setiap individu elemennya terdefinisi berdasarkan dua indeks (yang biasanya dikonotasikan dengan baris dan kolom).

Indeks pada matriks harus mempunyai batasan dan sifat yang sama dengan indeks pada array satu dimensi.

Sering dikatakan bahwa matriks adalah array dua dimensi, tapi patut diperhatikan bahwa pengertian berdimensi dua (baris dan kolom) adalah dalam pemikiran kita (penyajian logika).

Pengaturan letak elemen matriks dalam memori komputer selalu tetap sebagai deretan "sel yang linier". Pengertian dua dimensi ini hanya untuk mempermudah programmer dalam merancang programnya.

Secara umum dimensi dan banyaknya elemen Array/Matriks sebenarnya tidak dibatasi oleh Compiler, tetapi dibatasi oleh keterbatasan RAM yang disediakan.

Karena matriks menggunakan memory internal. Struktur data ini praktis untuk dipakai, namun memori yang dipakai besar.

Contoh matriks yang sering kita lihat sehari-hari adalah tabel.

Deklarasi array (Pseudocode) :

```
nama_var : array [rangeindeks1,rangeindeks2] of tipe
```

Deklarasi array (C++):

```
tipe nama_var[ukuran1][ukuran2];
```

Contoh :

Sebuah matrik A berukuran 2x3 dapat dideklarasikan sebagai berikut :

Algoritma :

a : array [1..2,1..3] of integer

a_{1,1} ← 11

$a_{1,2} \leftarrow 7$
 $a_{1,3} \leftarrow 4$
 $a_{2,1} \leftarrow 12$
 $a_{2,2} \leftarrow 3$
 $a_{2,3} \leftarrow 9$

C++:

```
int a[2][3] = {{11, 7, 4},{12, 3, 9}};
```

yang akan menempati lokasi memori dengan susunan berikut :

	0	1	2
0	11	7	4
1	12	3	9

Dan definisi variabel untuk setiap elemen tersebut adalah :

	0	1	2
0	a[0][0]	a[0][1]	a[0][2]
1	a[1][0]	a[1][1]	a[1][2]

Contoh Program 1 :

```

#include<iostream.h>
main()
{
  int A[3][3]={{1,2,3},{4,5,6},{7,8,9}};
  int i,j;

  for(i=0; i<3; i++)
  {
    for(j=0; j<3; j++)
      cout<<A[i][j]<<" ";
    cout<<endl;
  }
}

```

Output :

1 2 3
4 5 6
7 8 9

Contoh Program 2 :

```
#include<stdio.h>
void printArray(int [][][3]);
main()
{
    int matrik1 [2][3] = { {1, 2, 3}, {4, 5, 6}};
    int matrik2 [2][3] = { {1, 2, 3}, {4, 5}};
    int matrik3 [2][3] = { {1, 2}, {4} };
    printArray(matrik1);
    printArray(matrik2);
    printArray(matrik3);
    return 0;
}
void printArray(int a[][][3])
{   int i, j;

    for(i=0; i<=1; i++)
        {   for(j=0; j<=2; j++)
            printf("%d ", a[i][j]);
            printf("\n");
        }
}
```

Output :

```
1 2 3
4 5 6
1 2 3
4 5 0
1 2 0
4 0 0
```

Contoh Program 3 :

```
//program penjumlahan matrik
#include<iostream.h>
main()
{
    int A[3][2]={{1,2},{3,4},{5,6}};
    int B[3][2]={{7,8},{9,10},{11,12}};
    int C[3][2];
    int i,j;

    cout<<"Matrik A = \n";
    for(i=0; i<3 ; i++)
    {
```



```

        for(j=0; j<2 ; j++)
            cout<<A[i][j]<<" ";
        cout<<endl;
    }

    cout<<"Matrik B = \n";
    for(i=0; i<3 ; i++)
    {
        for(j=0; j<2 ; j++)
            cout<<B[i][j]<<" ";
        cout<<endl;
    }

    //penjumlahan matrix
    for(i=0; i<3 ; i++)
        for(j=0; j<2 ; j++)
            C[i][j]=A[i][j] + B[i][j];

    cout<<"Matrik C = \n";
    for(i=0; i<3 ; i++)
    {
        for(j=0; j<2 ; j++)
            cout<<C[i][j]<<" ";
        cout<<endl;
    }
}

```

D. Array Dimensi Banyak

Bentuknya dalam bahasa C++ :

```

tipe nama_var[ukuran1[ukuran2]...[ukuranN];

```

Contoh :

```

int data_huruf[2][4][6];

```

PENUTUP

Array merupakan suatu tipe data terstruktur. Deklarasi tipe ini tidak hanya dapat untuk data satu dimensi tetapi bisa sampai n dimensi. Array dua dimensi biasanya dipakai untuk perhitungan matriks dua dimensi.

SOAL-SOAL

Diketahui matriks A dan matriks B sebagai berikut :

$$A = \begin{pmatrix} 2 & 3 & 6 \\ 4 & 7 & 8 \end{pmatrix} \quad B = \begin{pmatrix} 3 & 5 \\ 2 & 4 \\ 8 & 6 \end{pmatrix}$$

Buatlah program untuk menghitung
matriks C = matriks A * matriks B