

Mengenai Matriks

IF2110/IF2111 – Algoritma dan Struktur Data
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Definisi

Matriks

- Sekumpulan informasi yang setiap individu elemennya terdefinisi berdasarkan dua buah indeks (yang biasanya dikonotasikan dengan baris dan kolom)
- Setiap elemen matriks dapat diakses secara langsung jika kedua indeks diketahui
- Indeksnya harus bertipe yang mempunyai keterurutan (sukcesor/predesesor), misalnya integer.

Matriks adalah struktur data dengan memori internal. Struktur ini praktis untuk dipakai tetapi memakan memori!

Matriks integer 100×100 memakan $10000 \times$ tempat penyimpanan integer.

Memori Matriks (1/2)

Matriks adalah struktur data statik (ukurannya ditentukan dari awal)

Untuk keperluan ini, sering kali memori dipesan “berlebihan” untuk kemudian dipakai sebagian saja

	1	2	3	4	5	6	7	8	9	10
1	1	1	1							
2	2	2	2							
3	3	3	3							
4	4	4	4							
5										
6										
7										
8										
9										
10										

Memori Matriks (2/2)

Struktur Matriks adalah struktur internal yang statis dan kontigu

Alokasi ukuran matriks yaitu $N \times M$ selalu dilakukan sekaligus. Dari $N \times M$, mungkin hanya sebagian yang digunakan, sehingga harus dibedakan antara:

- Definisi ruang memori seluruh matriks

- Memori yang efektif dipakai

Contoh-Contoh Matriks

matHari [1..7,1..3]: Nama hari ke-1 s.d. 7 dalam 3 bahasa (Indonesia, Inggris, Prancis)

	1 = INDONESIA	2 = INGGRIS	3 = PRANCIS
1	Senin	Monday	Lundi
2	Selasa	Tuesday	Mardi
3	Rabu	Wednesday	Mercredi
4	Kamis	Thursday	Jeudi
5	Jumat	Friday	Vendredi
6	Sabtu	Saturday	Samedi
7	Minggu	Sunday	Dimanche

Contoh-Contoh Matriks

$a[1..5,1..5]$: Matriks bilangan real

	1	2	3	4	5
1	12.1	7.0	8.9	0.7	6.6
2	0.0	1.6	2.1	45.9	55.0
3	6.1	8.0	0.0	3.1	21.9
4	9.0	1.0	2.7	22.1	6.2
5	5.0	0.8	0.8	2.0	8.1

Contoh-Contoh Matriks

matFrek ['A'..'E',1..7]: Matriks frekuensi kemunculan huruf 'A' s.d. 'E' pada 7 *body of text*

	1	2	3	4	5	6	7
'A'	12	71	82	0	62	30	11
'B'	0	1	2	45	5	3	10
'C'	6	8	0	3	21	3	6
'D'	9	1	2	22	6	9	7
'E'	5	0	0	2	8	45	23

Contoh-Contoh Matriks

matSurvey [1..7,1..4]: Matriks hasil survey pada titik kordinat. MatSurvey[i,j] adalah hasil pengukuran <temperatur, kecepatan angin> pada titik koordinat i, j.

	1	2	3	4
1	<24,5>	<24,5>	<30,5>	<25,5>
2	<23,56>	<3,6>	<40,5>	<2,2>
3	<22,73>	<7,3>	<60,6>	<8,3>
4	<21,56>	<8,5>	<9,8>	<7,4>
5	<23,56>	<12,50>	<3,36>	<30,6>
6	<20,0>	<2,56>	<5,46>	<20,99>
7	<30,0>	<9,0>	<15,0>	<27,0>

Contoh pemakaian matriks (1/2)

Matriks banyak digunakan dalam komputasi numerik untuk representasi dalam *finite element*.

Penggunaan matriks dalam matematika untuk perhitungan "biasa" terhadap matriks: penjumlahan, perkalian, menentukan determinan, menginvers sebuah matriks, *transpose*, dll.

Semua "perhitungan" itu menjadi tidak primitif, harus diprogram.

Contoh pemakaian matriks

Dalam perhitungan ilmiah di mana suatu sistem diwakili oleh matriks (elemen hingga dalam teknik sipil dan mesin).

Dalam persoalan pemrograman linier dan *operational research*.

Dalam persoalan algoritmik: untuk menyimpan informasi yang cirinya ditentukan oleh 2 komponen (yang nantinya diterjemahkan dalam baris dan kolom) dan diakses langsung. Contoh: merepresentasi "cell" pada sebuah *spreadsheet*, merepresentasi "ruangan" pada sebuah gedung bertingkat dan sebagainya.

Notasi algoritmik dari matriks

`namaMatriks[indeks1, indeks2]`

Domain:

Domain matriks sesuai dengan pendefinisian indeks

Domain isi matriks sesuai dengan jenis matriks

Cara mengacu melalui indeks: contoh:

`matHari[i,j]` jika `i` dan `j` terdefinisi

`matHari[1,7]`

`matSurvey[3,5]` untuk mengacu satu data survey

`matSurvey[3,5].temp` untuk mengacu data temperatur

Implementasi Fisik Matriks 1

Penggunaan type primitif matrix (jika tersedia)

Contoh:

```
matFrek: matrix ['A'..'E', 0..6] of integer
```

Sebuah matriks yang merepresentasi frekuensi huruf 'A' s.d. 'E', untuk 7 buah teks

```
a: matrix [0..4, 0..4] of real
```

Sebuah matriks seperti dalam matematika biasa

Implementasi Fisik Matriks 2

Penggunaan type array of array (array 2 dimensi)

Contoh:

```
matFrek: array ['A'..'E'] of array [0..6] of integer
```

Sebuah matriks yang merepresentasi frekuensi huruf 'A' s.d. 'E', untuk 7 buah teks

```
a: array [0..4] of array [0..4] of real
```

Sebuah matriks seperti dalam matematika ditulis sebagai: $a(i,j)$

ADT Matrix

IF2110/IF2111 – Algoritma dan Struktur Data
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Contoh ADT Matrix (eksplisit)

{ Ukuran maksimum baris dan kolom }

constant ROW_CAP: integer = 100

constant COL_CAP: integer = 100

{ Definisi Type Matrix dengan indeks integer }

type ElType: integer

type Matrix:

< mem: matrix[0..COL_CAP-1,0..ROW_CAP-1] of ElType,

rowEff: integer, *{ ukuran baris yg terdefinisi }*

colEff: integer *{ ukuran kolom yg terdefinisi }* >

{ Memori matriks yang dipakai selalu di “ujung kiri atas” }

Spesifikasi Operasi ADT Matrix

```
{ ***** DEFINISI PROTOTYPE PRIMITIF ***** }
```

```
{ *** Konstruktor membentuk Matrix *** }
```

```
procedure CreateMatrix(output m: Matrix, input nRows, nCols: integer)
```

```
{ Membentuk sebuah Matrix “kosong” berukuran nRows x nCols di “ujung kiri” memori }
```

```
{ I.S. nRows dan nCols adalah valid untuk memori matriks yang dibuat }
```

```
{ F.S. sebuah matriks m sesuai dengan def di atas terbentuk }
```


Spesifikasi Operasi ADT Matrix

```
{ *** Untuk sebuah matriks m yang terdefinisi: *** }  
function getRowEff(m: Matrix) → integer  
{ Mengirimkan jumlah baris Matriks m }  
function getColEff(m: Matrix) → integer  
{ Mengirimkan jumlah kolom Matriks m }  
function isMatrixIdxValid(i, j: integer) → boolean  
{ Mengirimkan true jika i, j adalah indeks yang valid sesuai kapasitas maksimum  
  sebuah Matrix, ROW_CAP dan COL_CAP }  
function isIdxEff (m: Matrix, i, j: integer) → boolean  
{ Mengirimkan true jika i, j adalah indeks efektif bagi m }  
function getElmt (m: Matrix, i, j: integer) → ElType  
{ Mengirimkan elemen m dg nomor baris i dan nomor kolom j }
```

Spesifikasi Operasi ADT Matrix

```
{ *** Operasi mengubah nilai elemen matriks: Set / Assign *** }  
procedure setRowEff (input/output m: Matrix, input nRows: integer)  
{ I.S. m sudah terdefinisi }  
{ F.S. Nilai m.rowEff diisi dengan nRows, }  
procedure setColEff (input/output m: Matrix, input nCols: integer)  
{ I.S. m sudah terdefinisi }  
{ F.S. Nilai m.colEff diisi dengan nCols }  
procedure setElmt (input/output m: Matrix, input i, j: integer, input x: ElType)  
{ I.S. m sudah terdefinisi }  
{ F.S. m(i,j) bernilai x }  
{ Proses: Mengisi m(i,j) dengan x }  
  
{ ***** Assignment Matrix ***** }  
procedure copyMatrix(input source: Matrix, output target: Matrix)  
{ Melakukan assignment target ← source }
```

Implementasi di C: Array 2 Dimensi (Statik)

Notasi Algoritmik

```
{ Deklarasi Array }  
nama_array: array [0..ncol-1] of  
             array [0..nrow-1] of  
             type_array
```

```
{ Cara mengacu elemen }  
nama_array[idx_col, idx_row]
```

```
{ Contoh: }  
mat: array [0..2] of  
      array [0..3] of integer  
mat[i,j] ← 9  
x ← mat[2,3]
```

Bahasa C

```
/* Deklarasi Array */  
type_array nama_array[ncol][nrow];
```

```
/* Cara mengacu elemen */  
nama_array[idx_col][idx_row]
```

```
/* Contoh: */  
int mat[3][4];  
  
mat[i][j] = 9;  
x = mat[2][3];
```

Latihan Soal ADT Matrix

IF2110/IF2111 – Algoritma dan Struktur Data
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Latihan 1

```
procedure readMatrix (output m: Matrix)
{ I.S. sembarang }
{ F.S. Ukuran efektif dan semua elemen m terdefinisi }
{ Proses: Membaca masukan ukuran baris dan kolom efektif
      dari keyboard dan melakukan CreateMatrix dan
      mengisi nilai efektifnya.
      Mengisi elemen m dengan pembacaan dari keyboard secara
      secara traversal per baris. }
```

Latihan 2

function isMatrixEqual(m1,m2: Matrix) → boolean

{ Mengirimkan true jika $m1 = m2$, yaitu ukuran efektif $m1$ dan $m2$ sama dan untuk setiap i,j yang merupakan indeks baris dan kolom, $m1[i,j]=m2[i,j]$ }

Latihan 3

function isSparse (m: matrix) → **boolean**

{ Mengirimkan true jika m adalah matriks sparse: matriks “jarang” dengan definisi: hanya maksimal 5% dari memori matriks yang efektif bukan bernilai 0 }

Latihan 4

function transpose (m: Matrix) → Matrix

{ Menghasilkan transpose dari m yaitu setiap elemen $m[i,j]$ ditukar nilainya dengan elemen $m[j,i]$ }