

# 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..ROW\_CAP-1,0..COL\_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..nrow-1] of  
             array [0..ncol-1] of type_array
```

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

```
{ 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[nrow][ncol];
```

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

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