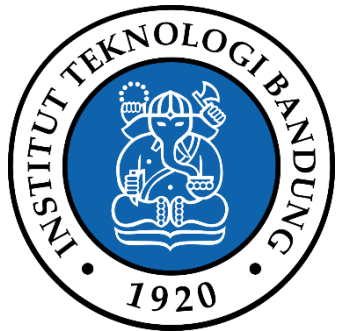
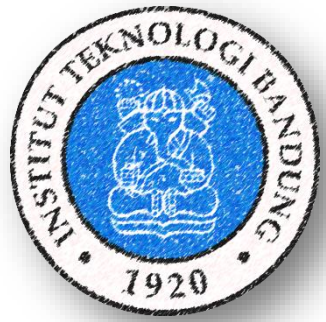


# Matriks (Python)

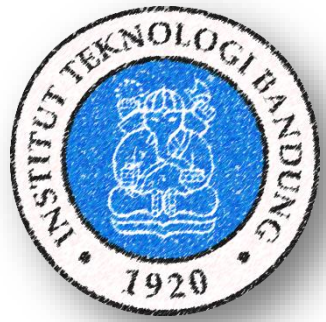
Tim Penyusun Materi Pengenalan Komputasi  
Institut Teknologi Bandung © 2019





# Tujuan

- Mahasiswa mampu:
  - Menjelaskan definisi dan kegunaan array
  - Menggunakan menggunakan notasi pendefinisian dan pengacuan array dengan benar
  - Melakukan pemrosesan sekuensial pada array



# 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.
  - Setiap elemen matriks mempunyai type yang homogen
  - Indeks baris dan kolom harus bertipe yang mempunyai keterurutan (suksesor/predesesor), misalnya integer.
- Matriks adalah struktur data dengan memori internal. Struktur ini praktis untuk dipakai tetapi memakan memori!
    - Matriks integer 100 x 100 memakan 10000 x tempat penyimpanan integer.

# Contoh-1

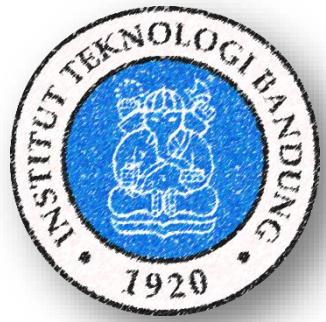
- MatUkur
  - Indeks (i, j) merepresentasikan suatu titik koordinat
  - Elemen matriks merepresentasikan hasil pengukuran pada suatu titik koordinat tertentu
  - Indeks baris : 1 s.d. 5, indeks kolom : 1 s.d. 5
  - Elemen matriks ber-type real

	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-2

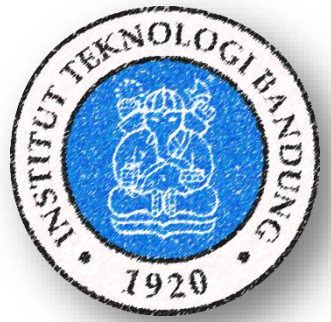
- MatSat
  - Merupakan matriks dengan elemen bernilai hanya 0 atau 1
  - Indeks baris : 1 s.d. 4; indeks kolom 1 s.d. 4
  - Elemen matriks ber-type integer

	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1



# Contoh Pemakaian

- Operasi “biasa” matriks dalam matematika : penjumlahan, perkalian, menentukan determinan, menginvers sebuah matriks, transpose, dll.
  - Semua "perhitungan" itu menjadi tidak primitif, harus diprogram
- Sistem persamaan linier dan *operational research*
- Persoalan algoritmik: untuk menyimpan informasi yang cirinya ditentukan oleh 2 dimensi (diterjemahkan dalam baris dan kolom).  
Contoh: *cell* pada sebuah *spreadsheet*, ruangan gedung bertingkat



# Implementasi Matriks di Python (1)

- Memori matriks diimplementasikan sebagai **array 2 dimensi**
- Suatu elemen matriks diakses dengan indeks baris dan kolom bertipe integer
- Elemen matriks dideklarasikan memiliki type yang sama (homogen)

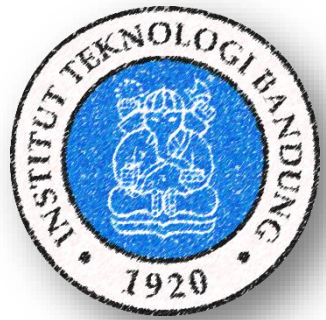
# Cara Deklarasi (1)

- Cara deklarasi sekaligus inisialisasi nilai matriks ukuran  $n \times m$ :

```
<nama-var> = [ [<val-11>, <val-12>, ..., <val-1m>] ,  
                [<val-11>, <val-12>, ..., <val-1m>] ,  
                ...  
                [<val-n1>, <val-n2>, ..., <val-nm>] ]
```

- Akan dideklarasikan array dengan ukuran sebesar  $n \times m$
- Type elemen tergantung pada nilai yang diberikan

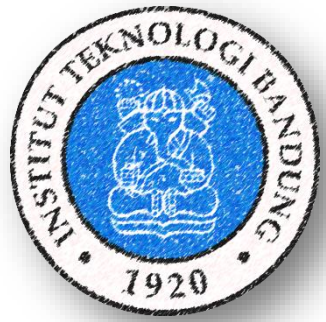




# Contoh Matriks-1

```
MatSatuan = [[0,1,0,1,1,1,0],  
              [1,1,1,1,0,1,1],  
              [0,0,0,1,1,0,1]]
```

Matriks bernama `MatSatuan` dengan setiap elemen bertipe `integer`, dengan `ukuran baris = 3` dan `ukuran kolom = 7`; dengan alamat setiap elemen diakses melalui `indeks baris 0 s.d. 2` dan `indeks kolom 0 s.d. 6`.

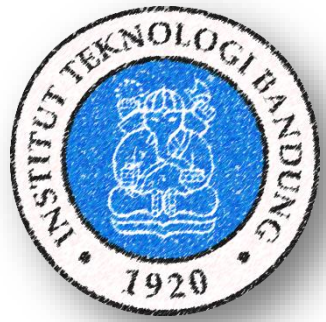


# Cara Deklarasi (2)

- Cara deklarasi:

**<nama-var> = [[<default-val> for j in range (<m>)] for i in range(<n>)]**

- Akan dideklarasikan array dengan ukuran n x m.
- Setiap elemen diberikan nilai 0.
- Type elemen integer



# Contoh Matriks 2

```
MatValue = [[0 for j in range (4)] for i in range(3)]
```

Matriks bernama **MatValue** dengan setiap elemen bertipe **integer**, dengan **ukuran baris = 3** dan **ukuran kolom = 4**; dengan alamat setiap elemen diakses melalui **indeks baris 0 s.d. 2** dan **indeks kolom 0 s.d. 3** dan setiap elemen diberi nilai **0**.

# Implementasi Matriks di Python (3)

- Cara akses elemen matriks:

`<namamatriks>[<nbrs>][<nkol>]`

- Contoh: `M1` dengan data sbb:

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	10
2	11	12	13	14	15
3	16	17	18	19	20
4	21	22	23	24	25

```
print (M1[4][0])
```

# akan tercetak: 21

```
x = M1[0][0] * M1[3][3]
```

# x bernilai 19

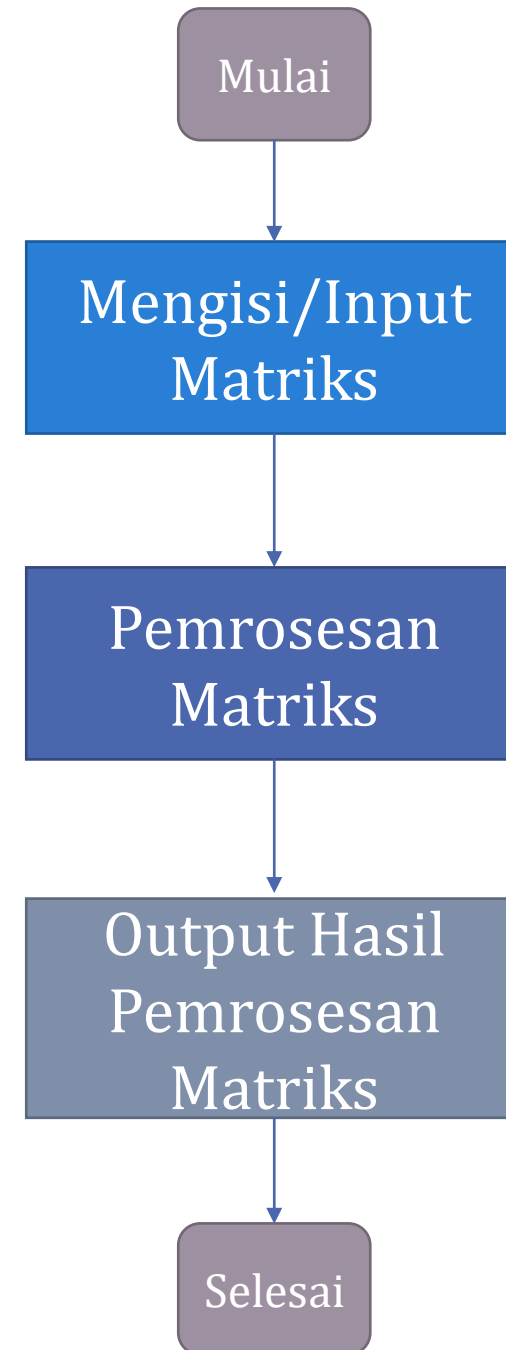
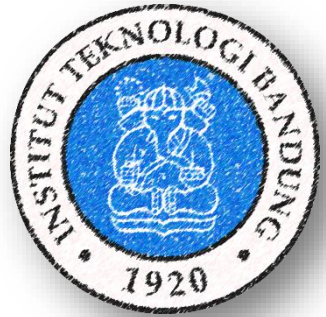
```
M1[1][1] = 8
```

# Elemen brs. 1, kol. 1 menjadi 8

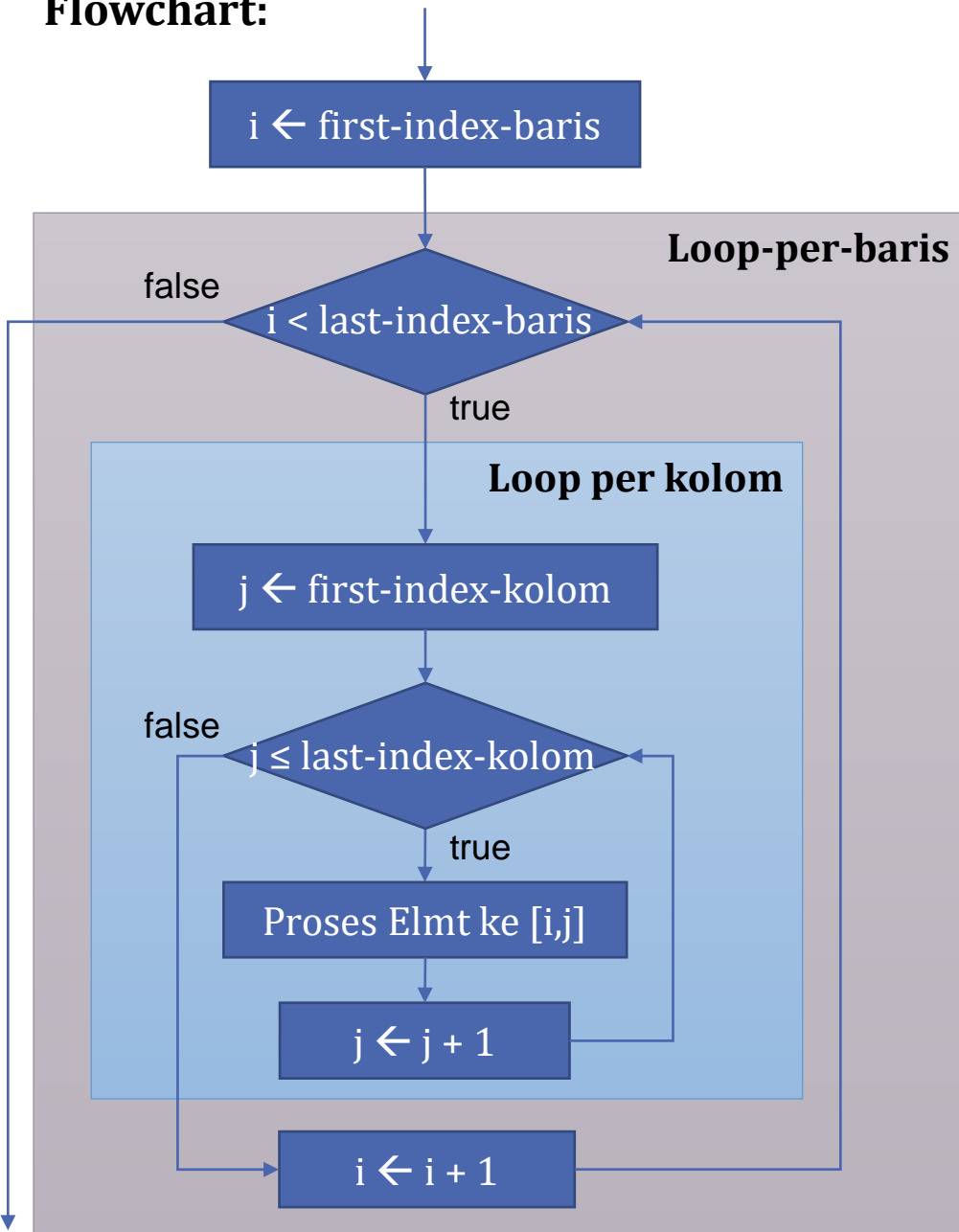
```
M1[0][5] ??
```

# **tidak terdefinisi! Tidak ada kolom dengan indeks = 5**

# Pemrosesan Matriks



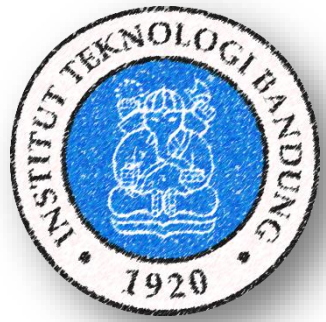
## Flowchart:



# Pemrosesan Dasar Matriks

## Pseudocode:

```
i traversal [first-index-baris..last-index-baris]
  j traversal [first-index-kolom..last-index-kolom]
    { Proses Element ke [i,j] }
    ...
```



# Operasi-Operasi pada Matriks

- Mendeklarasikan dan mendefinisikan isi matriks
- Menuliskan isi matriks ke layar
- Operasi 1 matriks:
  - Menghitung total semua elemen matriks
  - Mengalikan isi matriks dengan sebuah konstanta
  - Transpose matriks
- Operasi 2 matriks:
  - Menambahkan dua matriks
  - Mengalikan dua matriks

# Isi dan Tulis Matriks (1)

- Deklarasi matriks dan inisialisasi ukuran baris dan kolom
- Membaca isi matriks dari hasil kalkulasi
- Menampilkan ke layar

```
# Program IsiMatriks;
# Isi matriks dan menulis ke layar

# KAMUS
#   M : matriks of integer
#   NBrS, NKol : int (ukuran brs & kol)
#   i, j : int (indeks)

# ALGORITMA

# deklarasi matriks
NBrS = 5; NKol = 5;
M = [[0 for j in range(NKol)] for i in range(NBrS)]

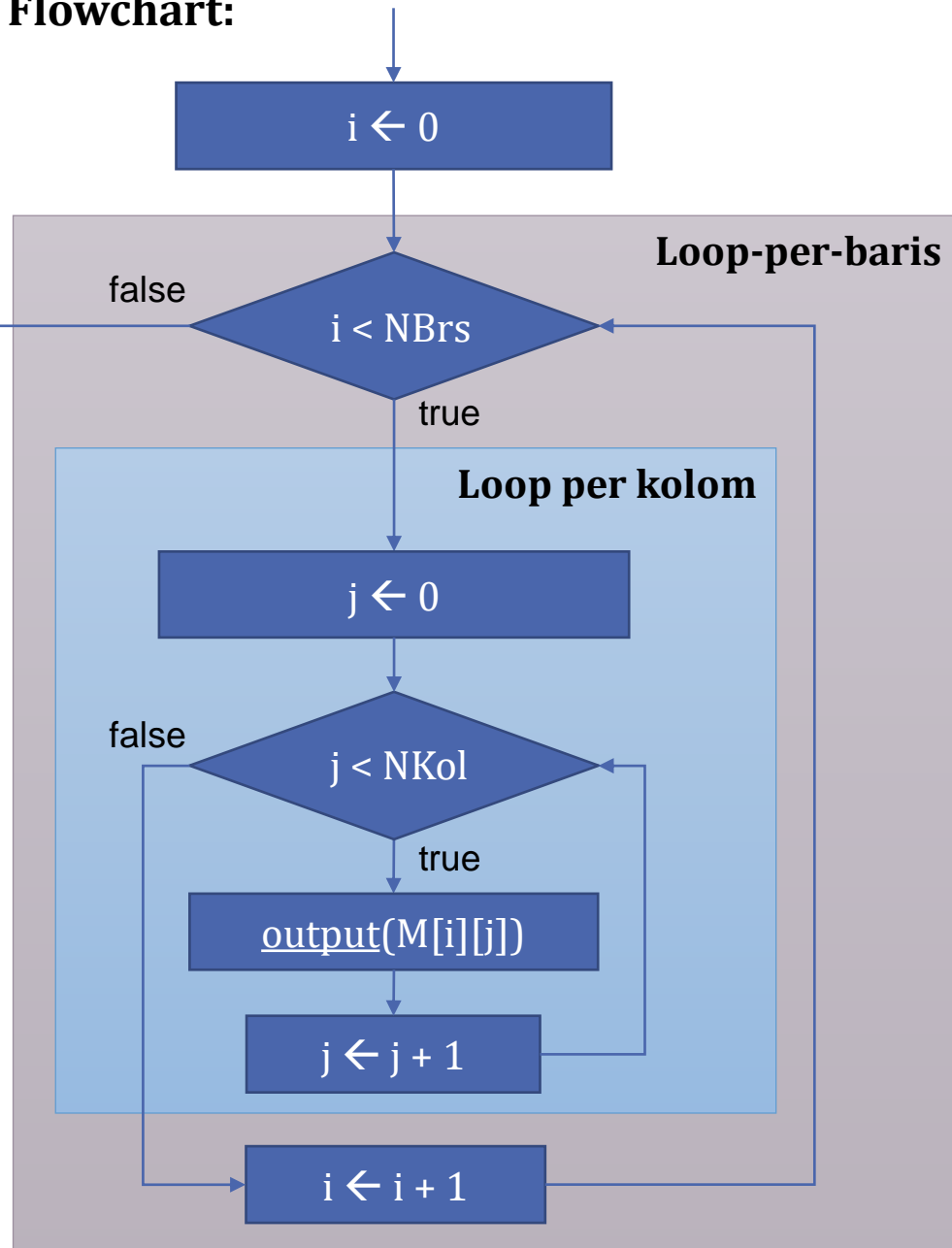
# Mengisi matriks ukuran NBrSxNKol
for i in range (NBrS):
    for j in range (NKol):
        M[i][j] = i * j

# Menuliskan isi matriks ke layar
for i in range (NBrS):
    for j in range (NKol):
        print(str(M[i][j])+" ", end='')
    print() # print hanya enter
```

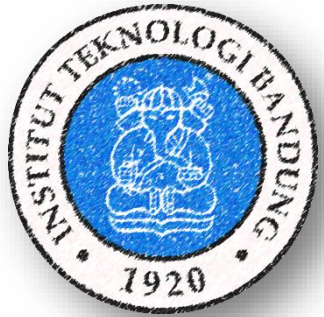
Print tanpa  
enter



## Flowchart:



# Isi dan Tulis Matriks (2) Bagian Tulis Matriks



## Pseudocode:

```
...  
{ Menuliskan isi matriks ke layar }  
i traversal [0..NBrS-1]  
  j traversal [0..NKol-1]  
    { Tulis Element ke [i,j] }  
    output(M[i][j])  
  ...
```

**Bagian kode yang lain silakan dibuat sebagai latihan**

## Note

- Untuk contoh-contoh selanjutnya, hanya disediakan kode program Python
- Silakan membuat *flowchart* dan *pseudocode* yang bersesuaian silakan dibuat sebagai latihan.

# Baca dan Tulis Matriks

- Deklarasi matriks
- Inisialisasi ukuran baris dan kolom dari masukan user
- Membaca isi matriks dari user
- Menampilkan isi matriks ke layar

```
# Program BacaMatriks;
# Baca isi matriks dari pengguna dan menulis ke layar
# KAMUS
#   M : matriks of integer
#   NBrS, NKol : int (ukuran brs & kol)
#   i, j : int (indeks)

# ALGORITMA

# deklarasi matriks
NBrS = 5; NKol = 5;
M = [[0 for j in range(NKol)] for i in range(NBrS)]

# Mengisi matriks ukuran NBrSxNKol
for i in range (NBrS):
    for j in range (NKol):
        M[i][j] = int(input("Elemen ke-["+str(i)+", "+str(j)+"] = "))

# Menuliskan isi matriks ke layar
for i in range (NBrS):
    for j in range (NKol):
        print(str(M[i][j])+" ", end='') # print tanpa enter
    print() # print hanya enter
```

# Sum Element Matriks

- Menjumlahkan seluruh elemen yang ada di matriks dan menampilkan hasilnya ke layar

```
# Program SumElmt
# Menjumlahkan isi elemen matriks ke layar
# KAMUS
#   M : matriks of integer
#   NBrS, NKol : int (ukuran brs & kol)
#   i, j : int (indeks)
#   sum : int (jumlah elemen)

# ALGORITMA

# deklarasi matriks
# Mengisi matriks ukuran NBrSxNKol
# Buat sebagai latihan

# Menjumlahkan elemen M
sum = 0
for i in range (NBrS):
    for j in range (NKol):
        sum = sum + M[i][j]

# Cetak nilai sum
print(sum)
```



# Latihan-1

- Buatlah program yang membaca sebuah matriks dengan elemen integer, misalnya M, dan masukan sebuah nilai integer, misalnya X dan selanjutnya mengalikan setiap elemen matriks M dengan X.
- Buatlah kode Python dan flowchart/pseudocode (sesuai arahan dosen kelas).

# Latihan-1

- Program Python

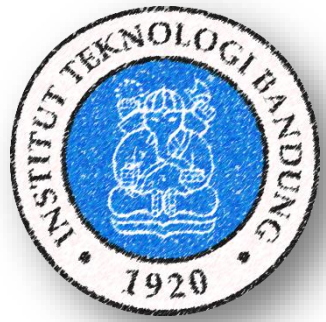
```
# Program KaliKons
# Mengalikan setiap elemen matriks dengan faktor pengali
# KAMUS
#   M : matriks of integer
#   NBrS, NKol : int (ukuran brs & kol)
#   i, j : int (indeks)
#   x : int (faktor elemen)

# ALGORITMA
# deklarasi matriks
# Mengisi matriks ukuran NBrSxNKol
# Buat sebagai latihan

# Input x
x = int(input("Faktor pengali = "))

# Menjumlahkan elemen M
for i in range (NBrS):
    for j in range (NKol):
        M[i][j] = M[i][j] * x

# Mencetak matriks baru ke layar
# Buat sebagai latihan
```



# Transpose Matriks (1)

- Dideklarasikan 2 buah matriks, misalnya M dan MTranspose
- MTranspose menampung hasil transpose dari M
  - Ukuran baris M = ukuran kolom MTranspose
  - Ukuran kolom M = ukuran baris Mtranspose
- $Mtranspose[i][j] = M[j][i]$

# Transpose Matriks (2)

- Program Python

```
# Program Transpose
# Transpose Matriks

# KAMUS
#   M : matriks of integer
#   NBrS, NKol : int (ukuran brs & kol)
#   MT : matriks of integer (matriks hasil transpose)
#   NBrST, NKolT : int (ukuran brs & kol hasil transpose)
#   i, j : int (indeks)

# ALGORITMA

# Deklarasi matriks M
# Mengisi matriks ukuran NBrSxNKol
# Buat sebagai latihan

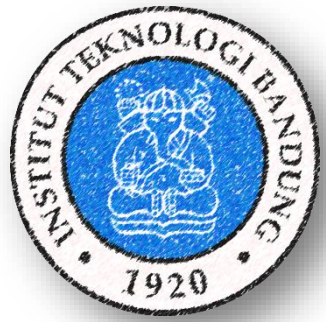
# Deklarasi MT
NBrST = NKol; NKolT = NBrS;
MT = [[0 for j in range(NKolT)] for i in range(NBrST)]

# Isi MT dengan hasil transpose M
for i in range (NBrST):
    for j in range (NKolT):
        MT[i][j] = M[j][i]

# Mencetak matriks MT ke layar
# Buat sebagai latihan
```







# Menjumlahkan 2 Matriks (1)

- Deklarasi 3 matriks, misal M1, M2, dan MHasil
- 2 matriks M1 dan M2 bisa dijumlahkan jika dimensinya sama, yaitu:
  - Ukuran baris M1 = ukuran baris M2
  - Ukuran kolom M1 = ukuran kolom M2
- Hasil ditampung di MHasil:  $MHasil[i][j] = M1[i][j] + M2[i][j]$
- Mengurangi M1 dengan M2 secara prinsip sama dengan menjumlahkan M1 dengan M2

# Menjumlahkan 2 Matriks (2)

## Program Python

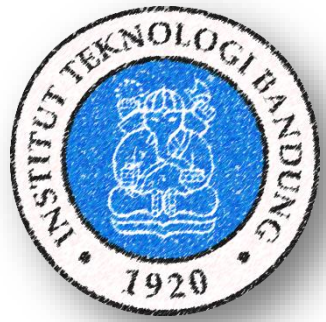
```
# Program Transpose
# Transpose Matriks
# KAMUS
#   M1, M2 : matriks of integer
#   NBrs1, NKol1 : int (ukuran brs & kol M1)
#   NBrs2, NKol2 : int (ukuran brs & kol M2)
#   MH : matriks of integer (matriks hasil penjumlahan)
#   NBrsH NKolH : int (ukuran brs & kol hasil penjumlahan)
#   i, j : int (indeks)

# ALGORITMA
# Deklarasi matriks M1 dan Mengisi matriks M1 ukuran NBrs1xNKol1
# Buat sebagai latihan1
# Deklarasi matriks M2 dan Mengisi matriks M2 ukuran NBrs2xNKol2
# Buat sebagai latihan

# Deklarasi MH
NBrsH = NBrs1; NKolH = NKol1;
MH = [[0 for j in range(NKolH)] for i in range(NBrsH)]

# Isi MH dengan hasil penjumlahan M1+M2
for i in range (NBrsH):
    for j in range (NKolH):
        MH[i][j] = M1[i][j] + M2[i][j]

# Mencetak MH ke layar
# Buat sebagai latihan
```



# Mengalikan 2 Matriks (1)

- Deklarasi 3 matriks, misal M1, M2, dan MHasil
- 2 matriks M1 dan M2 bisa dikalikan jika:
  - Ukuran kolom M1 = ukuran baris M2
- MHasil menampung hasil perkalian antara M1 dan M2
  - Ukuran baris MHasil = ukuran baris M1
  - Ukuran kolom MHasil = ukuran kolom M2

# Mengalikan 2 Matriks (2)

- Beberapa contoh:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} = \begin{pmatrix} a\alpha + b\gamma & a\beta + b\delta \\ c\alpha + d\gamma & c\beta + d\delta \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \begin{pmatrix} a & d \\ b & e \\ c & f \end{pmatrix} = \begin{pmatrix} 1a + 2b + 3c & 1d + 2e + 3f \\ 4a + 5b + 6c & 4d + 5e + 6f \\ 7a + 8b + 9c & 7d + 8e + 9f \end{pmatrix}$$

# Mengalikan 2 Matriks (3) Program Python

```
# Program KaliMatriks
# Perkalian matriks M1 dengan M2
# KAMUS
#   M1, M2 : matriks of integer
#   NBrs1, NKol1 : int (ukuran brs & kol M1)
#   NBrs2, NKol2 : int (ukuran brs & kol M2)
#   MH : matriks of integer (matriks hasil perkalian)
#   NBrsH NKolH : int (ukuran brs & kol hasil perkalian)
#   i, j, k : int (indeks)

# ALGORITMA
# Deklarasi matriks M1 dan Mengisi matriks M1 ukuran NBrs1xNKol1
# Buat sebagai latihan
# Deklarasi matriks M2 dan Mengisi matriks M2 ukuran NBrs2xNKol2
# Buat sebagai latihan

# Deklarasi MH
NBrsH = NBrs1; NKolH = NKol2;
MH = [[0 for j in range(NKolH)] for i in range(NBrsH)]

# Isi MH dengan hasil perkalian M1 dan M2
for i in range (NBrsH):
    for j in range (NKolH):
        MH[i][j] = 0
        for k in range (NKol1):
            MH[i][j] = MH[i][j] + (M1[i][k]*M2[k][j])

# Mencetak MH ke layar
# Buat sebagai latihan
```