

# Homework

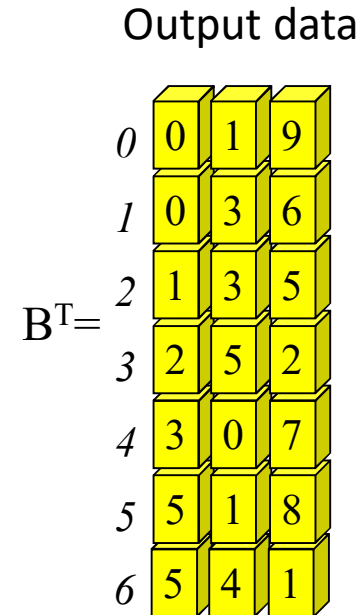
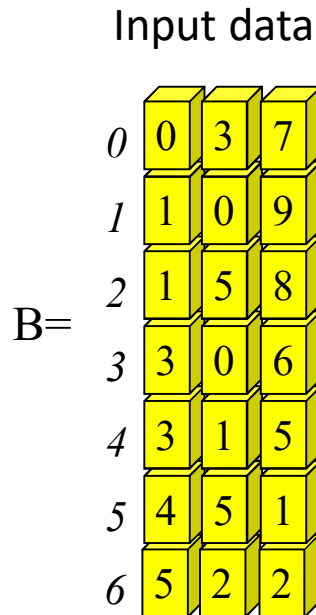
## 1. (Programming)

For a sparse matrix represented by storing only non-zero elements, implement a transpose operation.

e.g.  $A: m \times n$  matrix  $\rightarrow A^T: n \times m$  matrix

- Use the data structure in p26 of 'DS-Lec03-Array\_pointer.pdf'
- Sparse matrix is saved in a row-wise manner as below.
- Note that the transposed matrix  $B^T$  should also be saved in a row-wise manner.

$$B = \begin{bmatrix} 0 & 0 & 0 & 7 & 0 & 0 \\ 9 & 0 & 0 & 0 & 0 & 8 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 6 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 0 & 0 \end{bmatrix}$$



# Homework

```
void main ()
{
```

```
    //Add B as an input. Use your own example freely.
    SparseMatrix B = ;
```

$B =$

0	0	3	7
1	1	0	9
2	1	5	8
3	3	0	6
4	3	1	5
5	4	5	1
6	5	2	2

```
    //Perform the transpose operation
    .....
```

$B^T =$

0	0	1	9
1	0	3	6
2	1	3	5
3	2	5	2
4	3	0	7
5	5	1	8
6	5	4	1

```
    //Print out B and B^T in a dense matrix form to check
    //whether the operation works correctly.
```

```
    .....
```

$$B = \begin{bmatrix} 0 & 0 & 0 & 7 & 0 & 0 \\ 9 & 0 & 0 & 0 & 0 & 8 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 6 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 0 & 0 \end{bmatrix}$$

$$B^T = \begin{bmatrix} 0 & 9 & 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 1 & 0 \end{bmatrix}$$

# Homework

## 2. (Programming)

Implement a function 'mem\_alloc\_3D\_double' of allocating 3D array of double. Then, use this as below.

```
void main ()
{
    //Define two matrices A and B using 'mem_alloc_3D_double';
    double ***A = ;
    double ***B = ;

    //Perform addition of two matrices using 'addition_3D()'
    .....
    //Deallocate A and B
    .....
}
```