Arrays and Structures:

Matrix Transpose

Joseph Chuang-Chieh Lin (林莊傑)

Department of Computer Science & Engineering,
National Taiwan Ocean University

Fall 2024

Outline

Outline

Transposing a Matrix (1/4)

$M \in \mathbb{Z}^{6\times 6}$:

$$\begin{bmatrix} 15 & 0 & 0 & 22 & 0 & -15 \\ 0 & 11 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & -6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 91 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 28 & 0 & 0 & 0 \end{bmatrix}$$

Transposing a Matrix (1/4)

$M \in \mathbb{Z}^{6\times 6}$:

$$\begin{bmatrix} 15 & 0 & 0 & 22 & 0 & -15 \\ 0 & 11 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & -6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 91 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 28 & 0 & 0 & 0 \end{bmatrix}$$

$M^{\top} \in \mathbb{Z}^{6\times 6}$:

$$\begin{bmatrix} 15 & 0 & 0 & 0 & 91 & 0 \\ 0 & 11 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 28 \\ 22 & 0 & -6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -15 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Transposing a Matrix (2/4)

$M \in \mathbb{Z}^{6\times 6}$:

	Row	Col	Value
A[0]	6	6	8
A[1]	0	0	15
A[2]	0	3	22
A[3]	0	5	−15
A[4]	1	1	11
A[5]	1	2	3
A[6]	2	3	−6
A[7]	4	0	−91
A[8]	5	2	−28

Transposing a Matrix (2/4)

$M \in \mathbb{Z}^{6\times 6}$:

	Row	Col	Value
A[0]	6	6	8
A[1]	0	0	15
A[2]	0	3	22
A[3]	0	5	−15
A[4]	1	1	11
A[5]	1	2	3
A[6]	2	3	−6
A[7]	4	0	−91
A[8]	5	2	−28

$M^{\top} \in \mathbb{Z}^{6\times 6}$:

	Row	Col	Value
A[0]	6	6	8
A[1]	0	0	15
A[2]	0	4	91
A[3]	1	1	11
A[4]	2	1	3
A[5]	2	5	28
A[6]	3	0	22
A[7]	3	2	−6
A[8]	5	0	−15

Transposing a Matrix (3/4)

Algorithm 1

- for each row i,
  - place element $\langle i, j, \text{value} \rangle$ in element $\langle j, i, \text{value} \rangle$

Algorithm 2

- for each column j,
  - place element $\langle i, j, \text{value} \rangle$ in element $\langle j, i, \text{value} \rangle$

Transposing a Matrix (3/4)

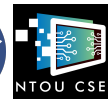
Algorithm 1

- for each row i,
  - place element $\langle i, j, \text{value} \rangle$ in element $\langle j, i, \text{value} \rangle$

Algorithm 2

- for each column j,
  - place element $\langle i, j, \text{value} \rangle$ in element $\langle j, i, \text{value} \rangle$

- What's the difficulty for Algorithm 1?

Transposing a Matrix (4/4) $O(\text{columns} \times \text{elements})$

```
void transpose(term a[], term b[]) { // b is set to the transpose of a
    int n, i, j, currentb;
    n = a[0].value; // total number of elements
    b[0].row = a[0].col; // rows in b = columns in a
    b[0].col = a[0].row; // columns in b = rows in a
    b[0].value = n;
    if (n > 0) { // dealing with a nonzero matrix
        currentb = 1;
        for (i=0; i<a[0].col; i++) // transpose by the columns in a
            for (j=1; j<=n; j++) // find elements from the current column
                if (a[j].col == i) { // element is in current column, add it to b
                    b[currentb].row = a[j].col;
                    b[currentb].col = a[j].row;
                    b[currentb].value = a[j].value;
                    currentb++;
                }
    }
}
```


Outline

Discussions