Lecture#4

Data Structures

Dr. Abu Nowshed Chy

Department of Computer Science and Engineering

University of Chittagong

January 19, 2025

Faculty Profile

2D Array

	Col. 0	Col. 1	Col. 2	Col. 3
Row. 0	(0, 0)	(0, 1)	(0, 2)	(0, 3)
Row. 1	(1, 0)	(1, 1)	(1, 2)	(1, 3)
Row. 2	(2, 0)	(2,1)	(2, 2)	(2, 3)
Row. 3	(3, 0)	(3, 1)	(3, 2)	(3, 3)

2D Array

The syntax for declaring a two-dimensional array is:

DataType ArrayName [RowSize][ColumnSize]

int data[3][4]

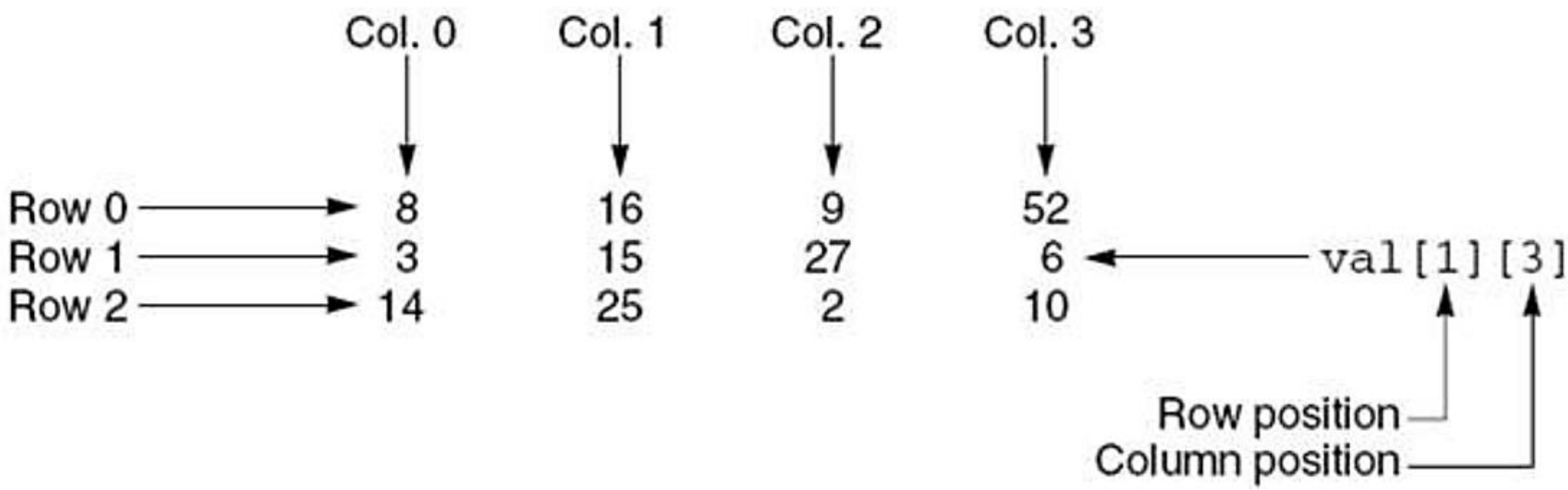
Number of elements = RowSize × ColumnSize

= 3 × 4

= 12

2D Array

Each array element is identified by its Row and Column position



2D Array

Assume 17 students had taken 4 class test.

The marks are stored in 17 × 4 array locations:

	CT0	CT1	CT2	CT3
Std 0	8	7	6	8
Std 1	0	7	8	9
Std 2	6	5	8	8
..	..	..	..	..
..	..	..	..	..
Std 16	7	8	8	7

Representation of 2D Array in Memory

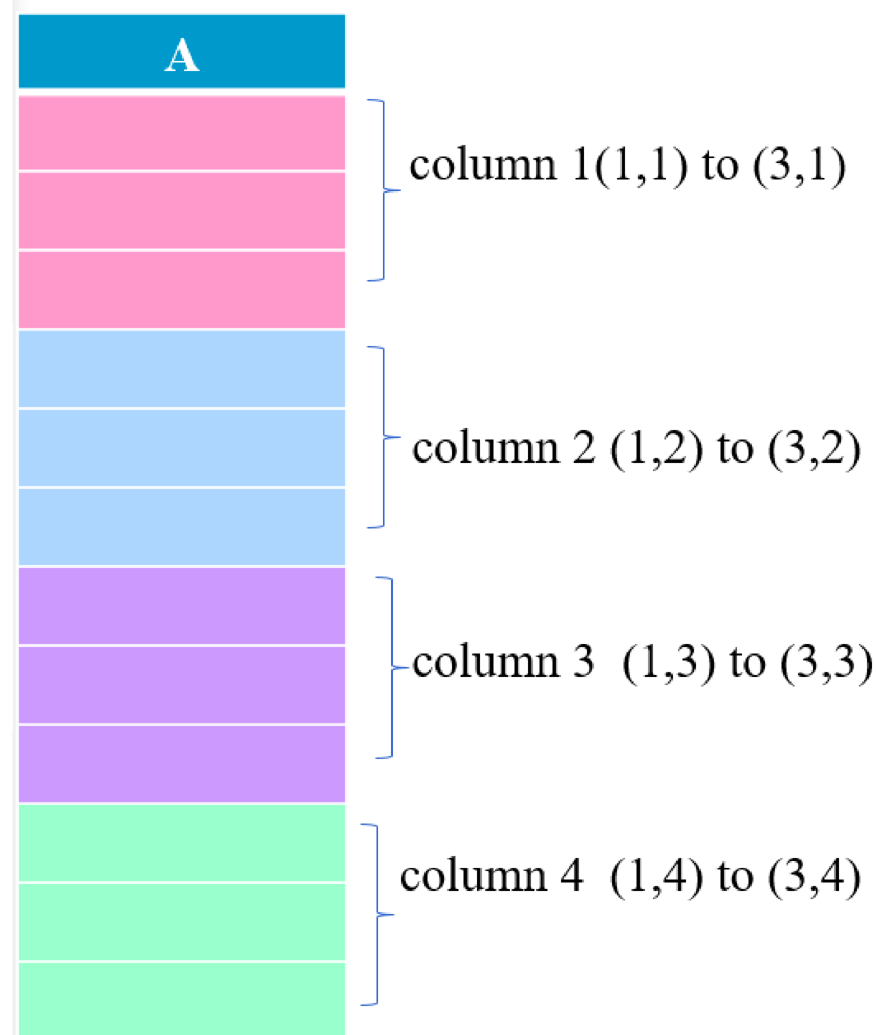
$$\begin{array}{cc}
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} [1,1] & [1,2] & [1,3] & [1,4] \\ [2,1] & [2,2] & [2,3] & [2,4] \\ [3,1] & [3,2] & [3,3] & [3,4] \end{bmatrix}
\end{array}$$

Two dimensional 3 x 4 array A

Representation of 2D Array in Memory

Column-major Order

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & [1,1] & [1,2] & [1,3] & [1,4] \\ 2 & [2,1] & [2,2] & [2,3] & [2,4] \\ 3 & [3,1] & [3,2] & [3,3] & [3,4] \end{array}$$



Representation of 2D Array in Memory

Column-major Order

w → Size of the data type. e.g. int → 4

$$LOC(A[J, K]) = \mathit{Base}(LA) + w[n(J-1) + (K-1)]$$

where

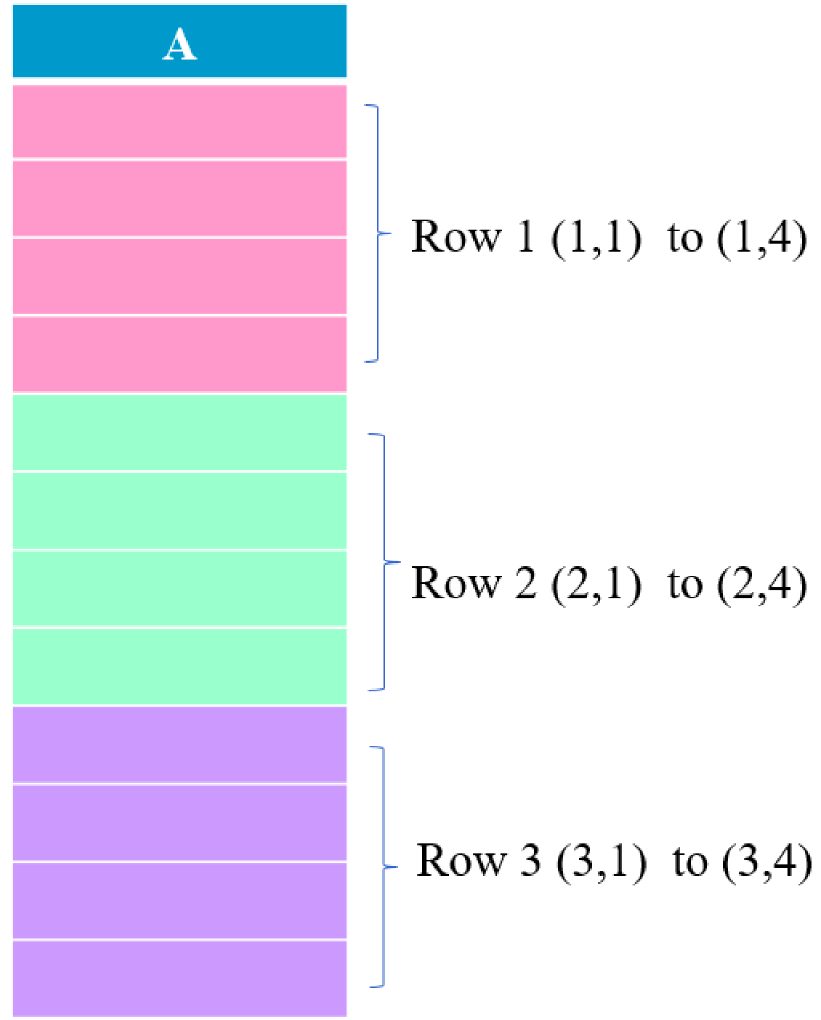
Base(A) = the address of the first element A[1,1] of A

n = a number of rows in one column

Representation of 2D Array in Memory

Row-major Order

$$
\begin{array}{c|cccc|}
 & 1 & 2 & 3 & 4 \\
1 & [1,1] & [1,2] & [1,3] & [1,4] \\
2 & [2,1] & [2,2] & [2,3] & [2,4] \\
3 & [3,1] & [3,2] & [3,3] & [3,4]
\end{array}
$$



Representation of 2D Array in Memory

Row-major Order

$$LOC(A[J, K]) = \mathit{Base}(LA) + w[n(J-1) + (K-1)]$$

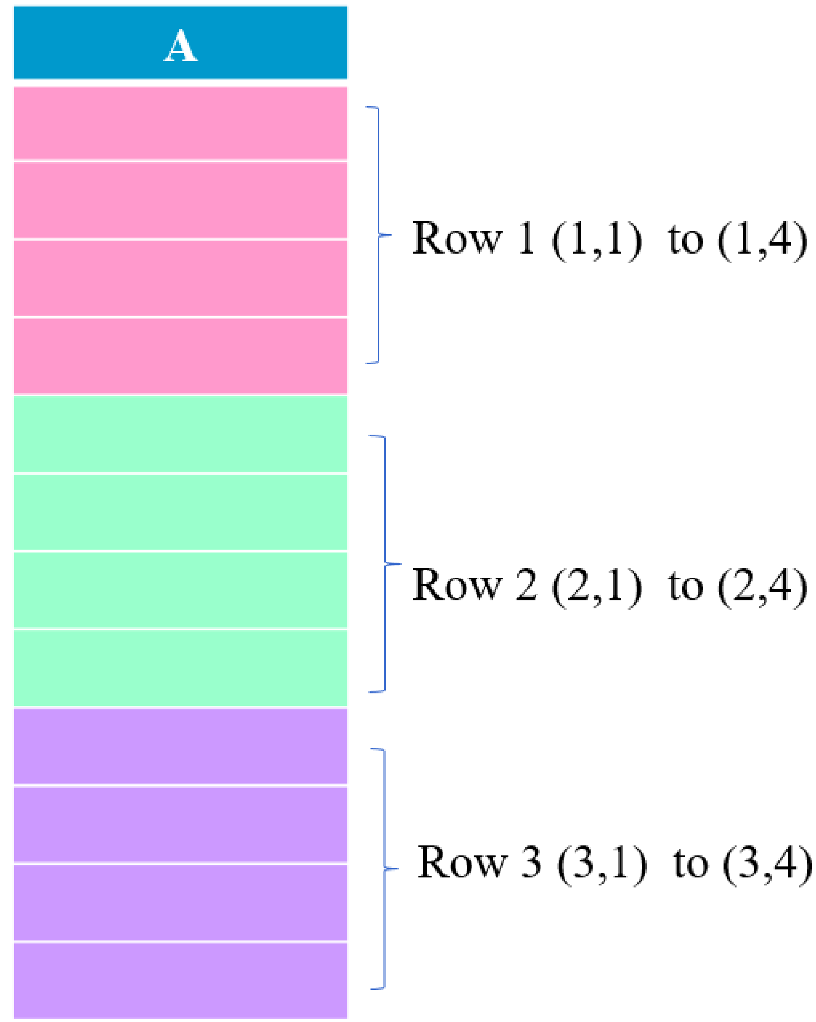
where

Base(A) = the address of the first element A[1,1] of A

n = a number of columns in one row

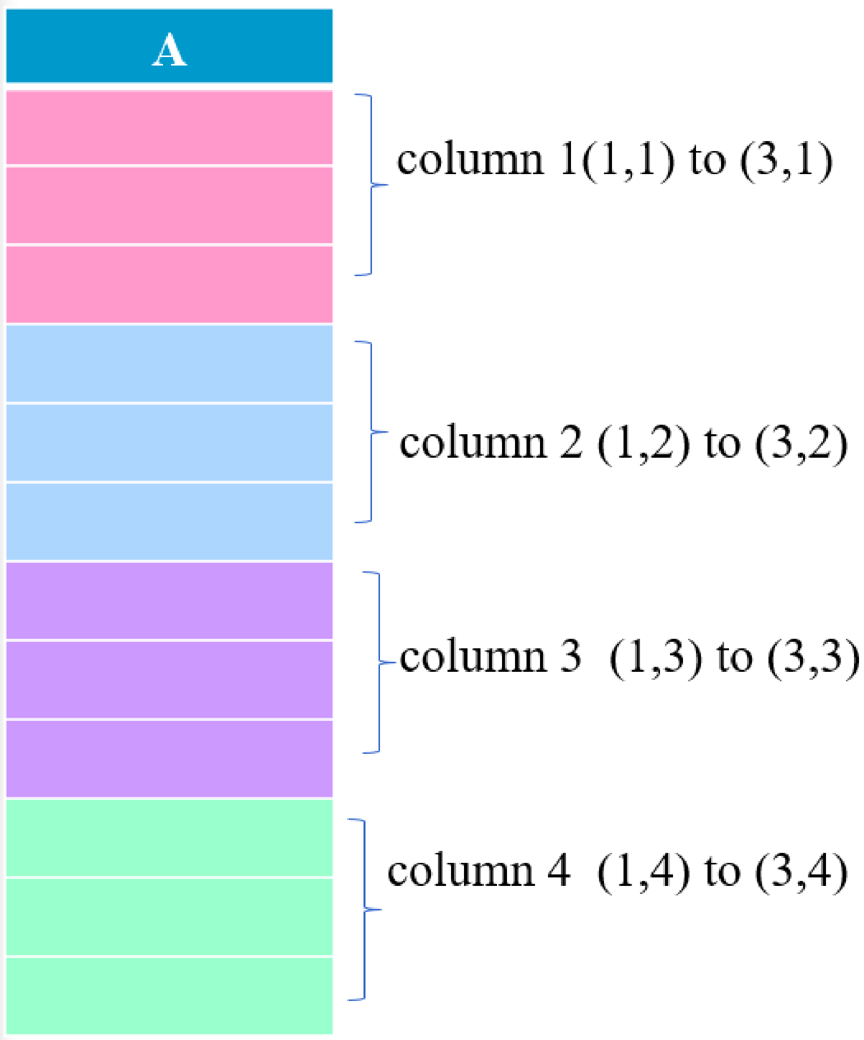
Representation of 2D Array in Memory

If *Base*(LA) = 200 and *w* = 4 what is the address of LOC(LA [3, 2]) in Row-major Order representation?



Representation of 2D Array in Memory

If *Base*(LA) = 200 and *w* = 4 what is the address of LOC(LA [3, 2]) in Column-major Order representation?



Longest Common Subsequence (LCS)

What is Subsequence?

A subsequence can be derived from the given sequence by deleting some or no elements without changing the order of the remaining elements.

The list of all subsequences for the word "apple" would be

"a", "ap", "al", "ae", "app", "apl", "ape", "ale", "appl", "appe", "aple", "apple", "p", "pp", "pl", "pe", "ppl", "ppe", "ple", "pple", "l", "le", "e", ""

Common Subsequences

Common subsequences are the subsequences that occur in both strings.

S1 = `abcde`

S2 = `cde`

Common Subsequences:

`c, d, e, cd, de, cde`

Longest Common Subsequence (LCS)

Given two sequences find the length of longest subsequence present in both items.

S1 = `abcde`

S2 = `cde`

Common Subsequences:

`c, d, e, cd, de, cde`

Longest Common Subsequence (LCS): `cde`

Longest Common Subsequence (LCS)

Given two sequences find the length of longest subsequence present in both items.

S1 = `x y z a b c d e f`

S2 = `z a d f`

Longest Common Subsequences: `z a d f`

Longest Common Subsequence (LCS)

Given two sequences find the length of longest subsequence present in both items.

S1 = `x y z a b c d e f`

S2 = `z y d b f`

Longest Common Subsequences: `zdf / ydf`

Longest Common Subsequence (LCS)

Given two sequences find the length of longest subsequence present in both items.

S1 = a b d a c e

S2 = b a b c e

Longest Common Subsequences?

Longest Common Subsequence Applications

- Similarity estimation of two strings or files
- Spoken word recognition
- Similarity of two biological sequences (DNA or protein)
- Sequence alignment
- In compressing genome resequencing data
- To authenticate users within their mobile phone through In-air signatures

Naïve /Brute Force Algorithm

For every subsequence of *X*, check whether it's a subsequence of *Y*.

- X has 2^m subsequences.
- Each subsequence takes O(*n*) time to check: scan *Y* for first letter, for second, and so on.

Time Complexity: $O(n2^m)$

Longest Common Subsequence (LCS)

We'll see how LCS algorithm works on the following example:

X = A B C B

Y = B D C A B

What is the Longest Common Subsequence (LCS) of X and Y?

Longest Common Subsequence (LCS)

The following steps are followed for finding the longest common subsequence.

1. Create a table of dimension `m+1*n+1` where `m` and `n` are the lengths of `X` and `Y` respectively.

Longest Common Subsequence (LCS)

X = ABCB;
m = |X| = 4

Y = BDCAB;
n = |Y| = 5

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi						
1	**A**						
2	**B**						
3	**C**						
4	**B**						

Longest Common Subsequence (LCS)

The following steps are followed for finding the longest common subsequence.

1. Create a table of dimension `m+1*n+1` where `m` and `n` are the lengths of `X` and `Y` respectively.
2. The first row and the first column are filled with zeros.

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**					
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

for i = 1 to m c[i,0] = 0
for j = 1 to n c[0,j] = 0

Longest Common Subsequence (LCS)

The following steps are followed for finding the longest common subsequence.

1. Create a table of dimension `m+1*n+1` where `m` and `n` are the lengths of `X` and `Y` respectively.
2. The first row and the first column are filled with zeros.
3. If the character corresponding to the current row and current column are matched, then fill the current cell by adding one to the diagonal element. Point an arrow to the diagonal cell.
4. Else take the maximum value from the previous column and previous row element for filling the current cell. Point an arrow to the cell with maximum value. If they are equal, point to any of them.

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**				
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**			
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**		
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**					
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	B	D	C	A	B
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**				
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**			
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**		
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**					
4	**B**	**0**					

Longest Common Subsequence (LCS)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	↑ **0**	↑ **0**	↑ **0**	↖ **1**	← **1**
2	**B**	**0**	↖ **1**	← **1**	← **1**	↑ **1**	↖ **2**
3	**C**	**0**	↑ **1**	↑ **1**	↖ **2**	← **2**	↑ **2**
4	**B**	**0**					

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	↑ **0**	↑ **0**	↑ **0**	↖ **1**	← **1**
2	**B**	**0**	↖ **1**	← **1**	← **1**	↑ **1**	↖ **2**
3	**C**	**0**	↑ **1**	↑ **1**	↖ **2**	← **2**	↑ **2**
4	**B**	**0**	↖ **1**	↑ **1**	↑ **2**	↑ **2**	↖ **3**

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

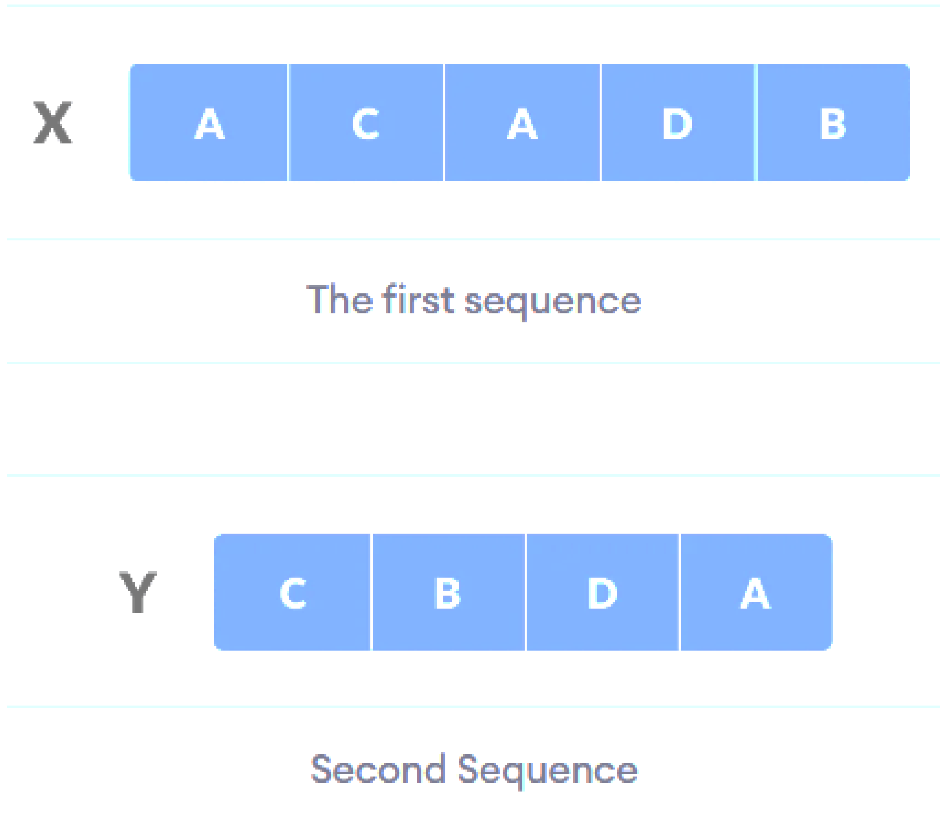
i	j	0	1	2	3	4	5
		Yj	**B**	**D**	**C**	**A**	**B**
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Longest Common Subsequence (LCS)

	j	0	1	2	3	4	5
i		Yj	B	D	C	A	B
0	Xi	**0**	**0**	**0**	**0**	**0**	**0**
1	**A**	**0**	**0**	**0**	**0**	**1**	**1**
2	**B**	**0**	**1**	**1**	**1**	**1**	**2**
3	**C**	**0**	**1**	**1**	**2**	**2**	**2**
4	**B**	**0**	**1**	**1**	**2**	**2**	**3**

Thus, the longest common subsequence is B C B

Longest Common Subsequence (LCS)



What is the Longest Common Subsequence (LCS) of X and Y?

Longest Common Subsequence (LCS)

		C	B	D	A
	0	0	0	0	0
A	0				
C	0				
A	0				
D	0				
B	0				

Longest Common Subsequence (LCS)

		C	B	D	A
	0	0	0	0	0
A	0	0	0	0	1
C	0				
A	0				
D	0				
B	0				

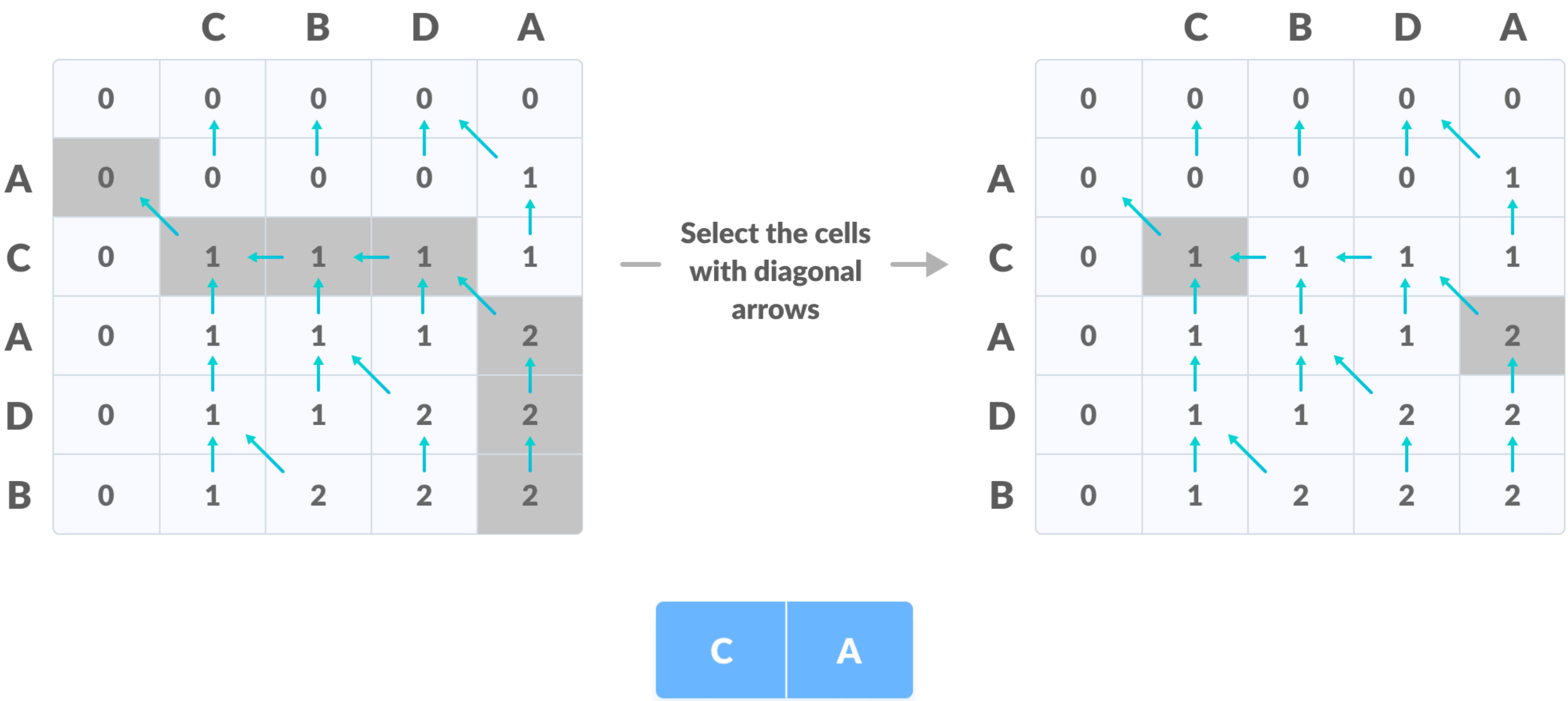
Longest Common Subsequence (LCS)

		C	B	D	A
	0	0	0	0	0
A	0	0	0	0	1
C	0	1	1	1	1
A	0	1	1	1	2
D	0	1	1	2	2
B	0	1	2	2	2

Longest Common Subsequence (LCS)

		C	B	D	A
	0	0	0	0	0
A	0	0	0	0	1
C	0	1	1	1	1
A	0	1	1	1	2
D	0	1	1	2	2
B	0	1	2	2	2

Longest Common Subsequence (LCS)



Longest Common Subsequence (LCS)

X = P R E S I D E N T

Y = P R O V I D E N C E

What is the Longest Common Subsequence of X and Y?

Longest Common Subsequence (LCS)

i	*j* *0*	*1* *p*	*2* *r*	*3* *o*	*4* *v*	*5* *i*	*6* *d*	*7* *e*	*8* *n*	*9* *c*	*10* *e*
0	0	0	0	0	0	0	0	0	0	0	0
1 *p*	0	↖ 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1
2 *r*	0	↑ 1	↖ 2	← 2	← 2	← 2	← 2	← 2	← 2	← 2	← 2
3 *e*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 2	↑ 2	↖ 3	← 3	← 3	↖ 3
4 *s*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 2	↑ 2	↑ 3	↑ 3	↑ 3	↑ 3
5 *i*	0	↑ 1	↑ 2	↑ 2	↑ 2	↖ 3	← 3	↑ 3	↑ 3	↑ 3	↑ 3
6 *d*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↖ 4	← 4	← 4	← 4	← 4
7 *e*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↖ 5	← 5	← 5	↖ 5
8 *n*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↑ 5	↖ 6	← 6	← 6
9 *t*	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↑ 5	↑ 6	↑ 6	↑ 6

Output: PRIDEN

Longest Common Subsequence (LCS)

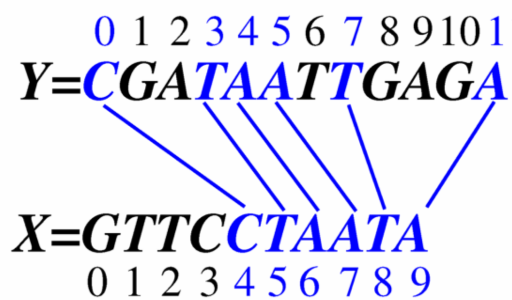
X = C G A T A A T T G A G A

Y = G T T C C T A A T A

What is the Longest Common Subsequence of X and Y?

Longest Common Subsequence (LCS)

L	-1	0	1	2	3	4	5	6	7	8	9	10	11
-1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	2	2	2	2	2	2	2	2	2
2	0	0	1	1	2	2	2	3	3	3	3	3	3
3	0	1	1	1	2	2	2	3	3	3	3	3	3
4	0	**1**	1	1	2	2	2	3	3	3	3	3	3
5	0	1	1	1	**2**	2	2	3	4	4	4	4	4
6	0	1	1	2	2	**3**	3	3	4	4	5	5	5
7	0	1	1	2	2	3	**4**	4	4	4	5	5	6
8	0	1	1	2	3	3	4	5	**5**	5	5	5	6
9	0	1	1	2	3	4	4	5	5	5	6	6	**6**



Longest Common Subsequence (LCS)

INPUT: two strings

X = A C T G A A C T C T G T G C A C T

Y = T G A C T C A G C A C A A A A A C

OUTPUT: longest common subsequence

A C T G A A C T C T G T G C A C T

T G A C T C A G C A C A A A A A C

Longest Common Subsequence (LCS)

Algorithm 7.1 LCS

Input: Two strings A and B of lengths n and m, respectively, over an alphabet Σ.

Output: The length of the longest common subsequence of A and B.

for $i \leftarrow 0$ **to** n
    $L[i, 0] \leftarrow 0$
end for
for $j \leftarrow 0$ **to** m
    $L[0, j] \leftarrow 0$
end for
for $i \leftarrow 1$ **to** n
    for $j \leftarrow 1$ **to** m
        if $a_i = b_j$ **then** $L[i, j] \leftarrow L[i-1, j-1] + 1$
        else $L[i, j] \leftarrow \max\{L[i, j-1], L[i-1, j]\}$
        end if
    end for
end for
return $L[n, m]$

Time Complexity: O(nm)


Thank You!