TOPIC ______________________ DATE ________

1. Solve the longest common subsequence (LCS) problem using dynamic programming.

- Provide DP table for inputs: X = {BCDAACD} and Y = {ACDBAC}
- Write the algorithm and calculate the time and space complexity.

ans: The longest common subsequence is a problem where one tries to find a group of alphabets, numbers which are common in both strings and are in same sequence in both strings. It doesn't matter if the letters are together or not.

X = {B C D A A C D}
is a 7 letter sequence
Y = {A C D B A C}
is a 6 letter sequence

Let us create a table of length 8, breadth 7 where the zeroth index will be present to help start the DP algorithm.

The algorithm is as follows: LCS-length (x, y, m, n).

```
let b[1:m, 1:n] and c[0:m, 0:n] be new tables
for i = 1 to m
    c[i,0] = 0
for j = 0 to n
    c[0,j] = 0
for i = 1 to m
    for j = 1 to n
        if x_i == y_j
            c[i,j] = c[i-1, j-1] + 1
            b[i,j] = "↖"
```

TOPIC ______ DATE ______

```
        else if c[i-1, j] ≥ c[i, j-1]
            b[i, j] = "↑"
            c[i, j] = c[i-1, j]

        else c[i, j] = c[i, j-1]
             b[i, j] = "←"

return c & b.

Print_LCS(b, X, i, j)
    if i==0 || j==0
        return
    if b[i, j] == "↖"
        Print_LCS(b, X, i-1, j-1)
        print x_i

    else if b[i, j] == "↑"
        Print_LCS(b, X, i-1, j)
    else
        Print_LCS(b, X, i, j-1)
```

DP table

			B	C	D	A	A	C	D
		0	1	2	3	4	5	6	7
	0	0	0	0	0	0	0	0	0
A	1	0	0	0	0	1	1	1	1
C	2	0	0	1	1	1	1	2	2
D	3	0	0	1	2	2	2	2	3
B	4	0	1	1	2	2	2	2	3
A	5	0	1	1	2	3	3	3	3
C	6	0	1	2	2	3	3	4	4

The longest common subsequence for the given strings is CDAC.

TOPIC ______ DATE ______

In the above algorithm everything depends on the 2D array/matrix formed. It is of size (len(x)+1, len(y)+1).
So, the time complexity depends on time taken to go across the matrix.
let us assume ~~to~~ the length is m.
let us assume the width is n.
So, time complexity is $O(m*n)$

In terms of space complexity.
Everything is stored inside the ~~m~~ matrix.
So, the space complexity is $O(m*n)$.

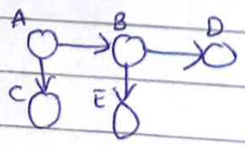
2. ~~Co~~ Consider the following directed graph.

$$\begin{tikzcd} A \arrow[r] \arrow[d] & B \arrow[r] \arrow[d] & D \\ C & E & \end{tikzcd}$$

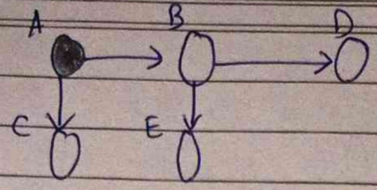
using the coloring scheme for graph traversal, simulate both BFS and DFS starting from vertex A.

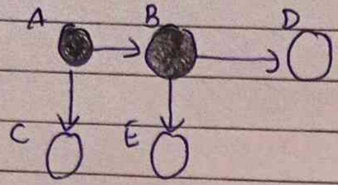
Colour meanings:
White: vertex has not been visited
Grey: vertex is ~~dis co~~ discovered but not explored
Black: vertex and all its neighbours are fully explored.

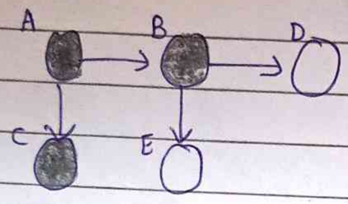
Ans: BFS (breadth first Search)

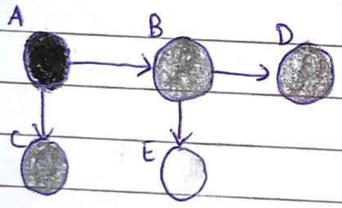


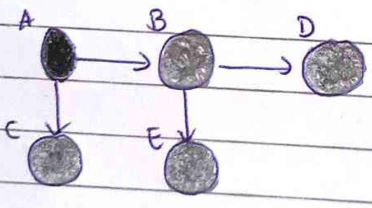
TOPIC ________ DATE ________

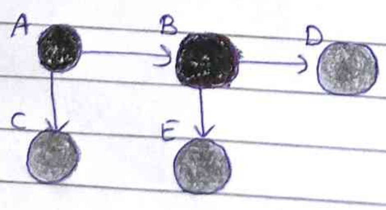




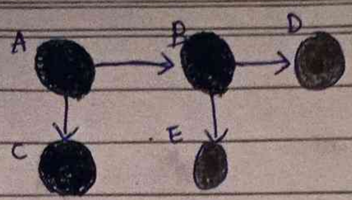


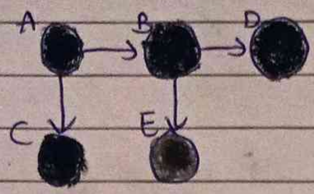


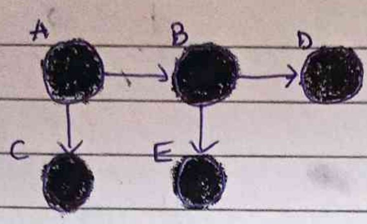




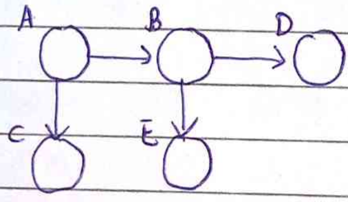
TOPIC

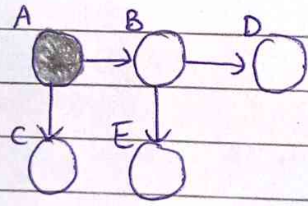


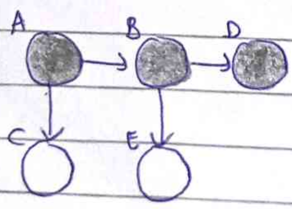


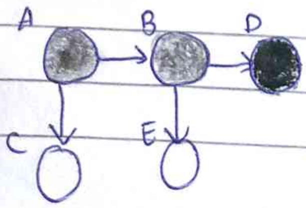


DFS (depth first search)

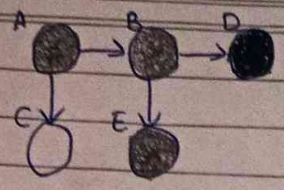


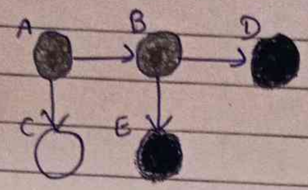


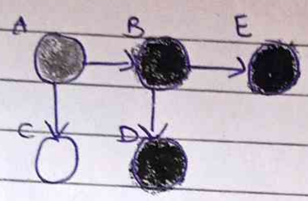


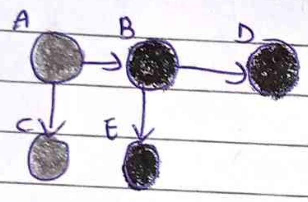


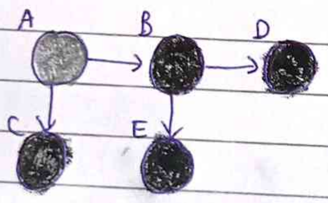
TOPIC DATE

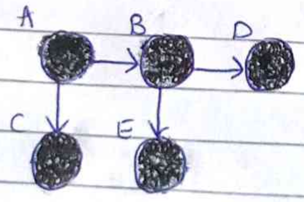












TOPIC ______________________ DATE ________

3. Find most efficient way to multiply these matrices using matrix chain multiplication.
Matrix sequence is ⟨4,10,3,12,20,7⟩.

Sol: The dimension of matrices from given matrix sequence is as follows:

$M_1 = 4 \times 10$
$M_2 = 10 \times 3$
$M_3 = 3 \times 12$
$M_4 = 12 \times 20$
$M_5 = 20 \times 7$

we can calculate the cost/no. of multiplication as follows:

$$c[i,j] = \min_{i \leq k < j} \{ \cancel{c[i]} \; c[i,k] + c[k+1,j] + d_{i-1} \times d_k \times d_j \}$$

to simplify our calculation we can use memoization, and use 2 tables as our bottom-up approach.

	1	2	3	4	5
1	0	120	264	1080	1344
2		0	360	1320	1350
3			0	720	1140
4				0	1680
5					0

cost table

	1	2	3	4	5
1		1	2	2	2
2			2	2	2
3				3	4
4					4
5					

K table (helps in finding parenthization)

∴ $c[1,1] = c[2,2] = c[3,3] = c[4,4] = c[5,5] = 0$
∵ a ~~multiply~~ matrix multiplying with no other matrix is ~~give~~ yielding zero cost.

TOPIC DATE

$\cancel{C[1,2] = \min_{1 \leq K < 2}^{K=1} \{C[1,1] + C[2,2] +}$

the d_n values are the dimensionality values given in the matrix sequence.

So, $d_0 = 4$
$d_1 = 10$
$d_2 = 3$
$d_3 = 12$
$d_4 = 20$
$d_5 = 7$

now,

$C[1,2] = \min_{1 \leq K < 2} \; K=1 \; \{C[1,1] + C[2,2] + d_0 \times d_1 \times d_2\}$

$C[1,2] = \{0 + 0 + 4 \times 10 \times 3\}$ [∵ there is only 1 possible k value hence the value will be min value itself]

$C[1,2] = 120$

$C[2,3] = \min_{2 \leq K < 3} \; K=2 \; \{C[2,2] + C[3,3] + d_1 \times d_2 \times d_3\}$

$= \{0 + 0 + 10 \times 3 \times 12\}$ [∵ itself the min value]

$= 360$

$C[3,4] = \min_{3 \leq K < 4} \; K=3 \; \{C[3,3] + C[4,4] + d_2 \times d_3 \times d_4\}$

$= \{0 + 0 + 3 \times 12 \times 20\}$ [∵ itself the min value]

$= 720$

TOPIC ______ DATE ______

$c[4,5] = \min_{4 \leq k < 5} {}^{k=4}\{c[4,4] + c[5,5] + 12 \times 20 \times 7\}$

$= \{0 + 0 + 1680\}$ [$\because$ itself the min value]

$= 1680$

$c[1,3] = \min_{1 \leq k < 3} \begin{matrix} k=1 \\ k=2 \end{matrix} \begin{Bmatrix} c[1,1] + c[2,3] + d_0 \times d_1 \times d_3 \\ c[1,2] + c[3,3] + d_0 \times d_2 \times d_3 \end{Bmatrix}$

$= \min \begin{matrix} k=1 \\ k=2 \end{matrix} \begin{Bmatrix} 0 + 360 + 4 \times 10 \times 12 \\ 120 + 0 + 4 \times 3 \times 12 \end{Bmatrix}$

$= \min \begin{matrix} k=1 \\ k=2 \end{matrix} \begin{Bmatrix} 840 \\ 264 \end{Bmatrix}$

$= 264$ at $k=2$

$c[2,4] = \min_{2 \leq k < 4} \begin{matrix} k=2 \\ k=3 \end{matrix} \begin{Bmatrix} c[2,2] + c[3,4] + d_1 \times d_2 \times d_4 \\ c[2,3] + c[4,4] + d_1 \times d_3 \times d_4 \end{Bmatrix}$

$= \min \begin{matrix} k=2 \\ k=3 \end{matrix} \begin{Bmatrix} 0 + 720 + 10 \times 3 \times 20 \\ 360 + 0 + 10 \times 12 \times 20 \end{Bmatrix}$

$= \min \begin{matrix} k=2 \\ k=3 \end{matrix} \begin{Bmatrix} 1320 \\ 2760 \end{Bmatrix}$

$c[2,4] = 1320$ [$k=2$]

$c[3,5] = \min_{3 \leq k < 5} \begin{matrix} k=3 \\ k=4 \end{matrix} \begin{Bmatrix} c[3,3] + c[4,5] + d_2 \times d_3 \times d_5 \\ c[3,4] + c[5,5] + d_2 \times d_4 \times d_5 \end{Bmatrix}$

$= \min \begin{matrix} k=3 \\ k=4 \end{matrix} \begin{Bmatrix} 0 + 1680 + 3 \times 12 \times 7 \\ 720 + 0 + 3 \times 20 \times 7 \end{Bmatrix}$

TOPIC DATE

$$= \min \begin{array}{l} K=3 \\ K=4 \end{array} \begin{Bmatrix} 1932 \\ 1140 \end{Bmatrix}$$

$$= 1140 (K=4)$$

$$C[1,4] = \min_{1 \le K < 4} \begin{array}{l} K=1 \\ K=2 \\ K=3 \end{array} \begin{Bmatrix} C[1,1] + C[2,4] + d_0 \times d_1 \times d_4 \\ C[1,2] + C[3,4] + d_0 \times d_2 \times d_4 \\ C[1,3] + C[4,4] + d_0 \times d_3 \times d_4 \end{Bmatrix}$$

$$C[1,4] = \min \begin{array}{l} K=1 \\ K=2 \\ K=3 \end{array} \begin{Bmatrix} 0 + 1320 + 4 \times 10 \times 20 \\ 120 + 720 + 4 \times 3 \times 20 \\ 264 + 0 + 4 \times 12 \times 20 \end{Bmatrix}$$

$$C[1,4] = \min \begin{array}{l} K=1 \\ K=2 \\ K=3 \end{array} \begin{Bmatrix} 2120 \\ 1080 \\ 1224 \end{Bmatrix}$$

$$C[1,4] = K=2 \{1080\}$$

$$C[2,5] = \min_{2 \le K < 5} \begin{array}{l} K=2 \\ K=3 \\ K=4 \end{array} \begin{Bmatrix} C[2,2] + C[3,5] + d_1 \times d_2 \times d_5 \\ C[2,3] + C[4,5] + d_1 \times d_3 \times d_5 \\ C[2,4] + C[5,5] + d_1 \times d_4 \times d_5 \end{Bmatrix}$$

$$= \min \begin{array}{l} K=2 \\ K=3 \\ K=4 \end{array} \begin{Bmatrix} 0 + 1140 + 10 \times 3 \times 7 \\ 360 + 1680 + 10 \times 12 \times 7 \\ 1320 + 0 + 10 \times 20 \times 7 \end{Bmatrix}$$

$$= \min \begin{array}{l} K=2 \\ K=3 \\ K=4 \end{array} \begin{Bmatrix} 1350 \\ 2880 \\ 2720 \end{Bmatrix}$$

$$C[2,5] = K=2 \{1350\}$$

TOPIC ____ DATE ____

$$C[1,5] = \min_{1 \le k < 5} \begin{cases} k=1 & C[1,1] + C[2,5] + d_0 \times d_1 \times d_5 \\ k=2 & C[1,2] + C[3,5] + d_0 \times d_2 \times d_5 \\ k=3 & C[1,3] + C[4,5] + d_0 \times d_3 \times d_5 \\ k=4 & C[1,4] + C[5,5] + d_0 \times d_4 \times d_5 \end{cases}$$

$$\text{or, } C[1,5] = \min \begin{cases} k=1 & 0 + 1350 + 4 \times 10 \times 7 \\ k=2 & 120 + 1140 + 4 \times 3 \times 7 \\ k=3 & 264 + 1680 + 4 \times 12 \times 7 \\ k=4 & 1080 + 0 + 4 \times 20 \times 7 \end{cases}$$

$$\text{or, } C[1,5] = \min \begin{cases} k=1 & 1630 \\ k=2 & 1344 \\ k=3 & 2280 \\ k=4 & 1640 \end{cases}$$

$$\text{or, } C[1,5] = \cancel{k=4\ \{1276\}}\ k=2\ \{1344\}$$

Now, analysing the table to create the proper parenthisation.

$$\cancel{(A(M_1 \times M_2) \times M_3 \times M_4) \times M_5}$$

$$(M_1) \times (M_2)) \times ((M_3) \times (M_4)) \times (M_5)$$

$$\Rightarrow (M_1 \cdot M_2) \times (M_3 \cdot M_4) \times M_5$$

TOPIC ______________________ DATE ______

4. Write an algorithm for merge Sort ? Sort following array elements using quicksort in ascending order
0, 8, 16, 5, 4, 9, 2, 7, 3

Sol: Merge Sort is a type of recursive sorting algorithm that works on the concept of divide and conquer. Where a big problem is repeatedly broken down into smaller problems. ~~Who~~ Then they are solved and all the smaller problems combine in such a way that they provide a solution to the original bigger problem.

The algorithm of mergesort is as follows:

```
MergeSort (arr, l, h){
    if (l<h)
     mid = (l+h)/2;
     MergeSort (arr, l, mid);
     MergeSort (arr, mid+1, h);
     Merge (arr, l, mid, h);
}
Merge (arr, l, mid, h){
        temp=[ ];
        i = l;
        j = mid+1;
    while (i<=mid && j<=h){
     if (arr[i]< arr[j])
         add (temp, arr[i]) // appends arr[i] in temp
     else
         add (temp, arr[j]) // appends arr[j] in temp
    }
```

TOPIC

DATE

// to add remaining elements of list.

~~if ( )~~

if (i > mid)

copy (all elements in the other list to temporary array)

else

copy (all elements in the other list to temporary array)

copy (temp, arr) // copy temp array to main array

Quick Sort

0 8 1 6 5 4 9 2 7 3

Pivot = 0

0 8 1 6 5 4 9 2 7 3

Pivot = 8

0 7 1 6 5 4 3 2 8 9

Pivot = 7

0 2 1 6 5 4 3 7 8 9

Pivot = 2

0 1 2 6 5 4 3 7 8 9

Pivot = 6

0 1 2 3 5 4 6 7 8 9

Pivot = 5

0 1 2 3 4 5 6 7 8 9