Assignment 3

Group member 1:-
Samad Amir (21K-3873)
Group member 2:-
Sandesh Goindani (21K-3898)
Group member 3:-
Umer Khan (21K-3912)

a)

Let

X : BSCABA
Y : ABCBDAB

j		0	1	2	3	4	5	6
i	Y \ X	O	B	S	C	A	B	A
0	O	0	0	0	0	0	0	0
1	A	0	↑0	↖0	↖0	↖1	↖1	↖1
2	B	0	↖1	↖1	↖1	↖1	↖2	↖2
3	C	0	↑1	↖1	↖2	↖2	↖2	↖2
4	B	0	↖1	↖1	↑2	↑2	↖3	↖3
5	D	0	↑1	↖1	↑2	↖2	↑3	↖3
6	A	0	↑1	↖1	↑2	↖3	↖3	↖4
7	B	0	↖1	↖1	↑2	↑3	↖4	↑4

① if $(X[i] == Y[j])$; then
$c[i][j] = 1 + c[i-1][j-1]$; ↖

② $c[i][j] = \max\left(c[i][j-1],\ c[i-1],[j]\right)$

$\because$ for $C[2][2]$ $\because X[i] = Y[j]$

$c[i][j] = 1 + c[1][1]$
$= 1 + 0$
$= 1$.

for $C[3,4]$
$= \max(c[3][3], c[2][4])$
$= \max(2, 1)$
$= 2$.

Length of LCS is : 4

and LCS is . BCBA , BCAB

b)

Member 1   Member 2

x: Samad   Y: Sandesh

Step 01:- Find LCS

j	x\Y	0	S	A	N	D	E	S	H
0	0	0	0	0	0	0	0	0	0
1	S	0	↖1	←1	←1	←1	←1	←1	←1
2	A	0	↑1	↖2	←2	←2	←2	←2	←2
3	M	0	↑1	↑2	←2	←2	←2	←2	←2
4	A	0	↑1	↖2	←2	←2	←2	←2	←2
5	D	0	↑1	↑2	←2	↖3	←3	←3	←3

LCS Length = 3

LCS = SAD

For length of SCS : length of (x) + Length of (Y) - LCS length

= 5 + 7 - 3 = 9

S A M A D

S A N D E S H

∴ If string matches write one time

$\boxed{\text{SCS = SAMNADESH}}$

Date:

c)

Select 4^{th} digit

Member 1: Samad = a = 1.

Member 2: Sandesh = d = 4

- Make array of temp and length of the orignal size array; fill length array with 1

L = [1, 10, 2, 4, 20]

- Initialize i = 1, j = 0.

- Update j til i; when j reaches i increment i and update j = 0.

- if j < i then add its index value and length + 1

Length	1	2	2	3	~~3~~4	~~4~~
Temp		0	0	2	3	~~3~~

Max length = 4

Date:

d)

str_1 = E L E P H A N T

str_2 = R E L E V A N T

C	0	R	E	L	E	V	A	N	T
0	0	1	2	3	4	5	6	7	8
E	1	1	~~0~~1	2	~~0~~3	4	5	6	7
L	2	2	2	1	2	3	4	5	6
E	3	3	2	2	1	2	3	4	5
P	4	4	3	3	2	2	3	4	5
H	5	5	4	4	3	3	3	4	5
A	6	6	5	5	4	4	3	4	5
N	7	7	6	6	5	5	4	3	4
T	8	8	7	7	6	6	5	4	3

Edit distance = 3

Replace H => V, Delete P at 4, Insert R at 1

Elephant -> Relevant

if ($s[i] == s_2[j]$)

$T[i,j] = T[i-1][j-1]$

else

$T[i][j] = \min(T[i-1][j], T[i][j-1], T[i-1][j-1])$

Date:

e)

Member 3: Umer = length = 4

$\underset{2\times25}{A_1}$ $\underset{25\times3}{A_2}$ $\underset{3\times16}{A_3}$ $\underset{16\times1}{A_4}$ $\underset{1\times4}{A_5}$

$p_0 = 2,\ p_1 = 25,\ p_2 = 3,\ p_3 = 16,\ p_4 = 1,\ p_5 = 4$

$m[i,j] = \min(m[i,k] + m[k+1,j] + p_{i-1} \times p_k \times p_j);$

$k=1$

$m[1,2] = \min[m[1,1] + m[2,2] + 2\times25\times3]$

$m[1,2] = 150$

$k=2$

$m[2,3] = [m[2,2] + m[3,3] + p_1 \times p_2 \times p_3]$

$= [0 + 0 + 25\times3\times16]$

$m[2,3] = 1200$

m	1	2	3	4	5
1	0	150	300	204	212
2		0	1200	447	360
3			0	48	60
4				0	64
5					0

$k=3$

$m[3,4] = [m[3,3] + m[4,4] + p_2 \times p_3 \times p_4]$

$= [0 + 0 + 3\times16\times1]$

$m[3,4] = 48$

$k=4$

$m[4,5] = [m[4,4] + m[5,5] + p_3 \times p_4 \times p_5]$

$= [0 + 0 + 16\times1\times4]$

$m[4,5] = 64$

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

$k = 1, k = 2$

$$m[1,3] = \min \begin{cases} k=1 & m[1,1] + m[2,3] + p_0 \times p_1 \times p_3 \\ k=2 & m[1,2] + m[3,3] + p_0 \times p_2 \times p_3 \end{cases}$$

Date:

$= \min(0 + 1200 + 2 \times 25 \times 16,\ 150 + 0 + 2 \times 3 \times 25)$

$= \min(2000, 300)$

$= 300 \ ; k = 2$

$k = 2, k = 3$

$$m[2,4] = \min_{\substack{k=2 \\ k=3}} \begin{cases} m[2,2] + m[3,4] + p_1 \times p_2 \times p_4 \\ m[2,3] + m[4,4] + p_1 \times p_3 \times p_4 \end{cases}$$

$= \min(0 + 1200 + 25 \times 3 \times 1,\ 48 + 0 + 25 \times 16 \times 1)$

$= \min(1275, 448) \ ; k = 3 = 448$

$$m[3,5] = \min_{\substack{k=3 \\ k=4}} \begin{cases} m[3,3] + m[4,5] + p_2 \times p_3 \times p_5 \\ m[3,4] + m[5,5] + p_2 \times p_4 \times p_5 \end{cases}$$

$= \min(0 + 64 + 3 \times 16 \times 4,\ 48 + 0 + 3 \times 1 \times 4)$

$= \min(256, 60)$

$= 60$

$$m[1,4] = \min_{\substack{k=1 \\ k=2 \\ k=3}} \begin{cases} m[1,1] + m[2,4] + p_0 \times p_1 \times p_4 \\ m[1,2] + m[3,4] + p_0 \times p_2 \times p_4 \\ m[1,3] + m[4,4] + p_0 \times p_3 \times p_4 \end{cases}$$

$= \min(0 + 448 + 2 \times 25 \times 1,\ 150 + 48 + 2 \times 3 \times 1,\ 300 + 0 + 2 \times 16 \times 1)$

$= \min(498, 204, 332)$

$= 204 \ ; k = 2$

$$m[2,5] = \min_{\substack{k=2 \\ k=3 \\ k=4}} \begin{cases} m[2,2] + m[3,5] + p_1 \times p_2 \times p_5 \\ m[2,3] + m[4,5] + p_1 \times p_3 \times p_5 \\ m[2,4] + m[5,5] + p_1 \times p_4 \times p_5 \end{cases}$$

$= \min(0 + 60 + 25 \times 3 \times 4,\ 1200 + 64 + 2 \times 16 \times 4,\ 448 + 0 + 25 \times 1 \times 4)$

$\min(360, 392, 548) = 360$

$k = 1$

Date:

$$m[1,5] = \min \begin{cases} k=1 & m[1,1] + m[2,5] + p_0 \times p_1 \times p_5 \\ k=2 & m[1,2] + m[3,5] + p_0 \times p_2 \times p_5 \\ k=3 & m[1,3] + m[4,5] + p_0 \times p_3 \times p_5 \\ k=4 & m[1,4] + m[5,5] + p_0 \times p_4 \times p_5 \end{cases}$$

$$= \min \begin{pmatrix} 0 + 360 + 2 \times 25 \times 4,\ 150 + 60 + 2 \times 3 \times 4,\ 300 + 64 + 2 \times 16 \times 4, \\ 204 + 0 + 2 \times 1 \times 4 \end{pmatrix}$$

$$= \min(560, 234, 462, 212)$$

$$= 212 \quad K = 4$$

Paren thesis

$((A_1 . A_2).(A_3 . A_4)).A_5$

(f) 0-1 Knapsack Problem

$i = \{1, 2, 3, 4, 5\}$

$P = \{1, 4, 5, 7, 4\}$

$W = \{1, 3, 4, 5, 2\}$

$m = 9$

Table V

P_i	w_i	item no (i)	(w) capacity									
			0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	1	1	1	1	1	1	1	1	1
4	3	2	0	1	1	4	5	5	5	5	5	5
5	4	3	0	1	1	4	5	6	6	9	10	10
7	5	4	0	1	1	4	5	7	8	9	11	12
4	2	5	0	1	4	5	5	8	9	11	12	13

Now, we check whether to add <u>item 5</u>:

$$V[i, w] = \max\{V[i-1, w],\ V[i-1, w - w(i)] + P[i]\}$$

i = item no
w = bag capacity

$$V[5, 9] = \max\{V[4, 9],\ V[4, 9-2] + 4\}$$

$\Rightarrow$ $V[5,9] = \max\{12, 13\}$

- We pick 5^{th} item, since the condition $V[i,w] = V[i-1, w]$ is false.

- Reduce bag capacity:

$W = 9 - 2$

$\boxed{W = 7}$

- ∴ Max profit (i) − profit (i)

$\Rightarrow$ $13 - 4$

$\Rightarrow$ $\boxed{9}$ → we will backtrack to see where 9 lies in the table and at which item

- <u>Now, we check for item 4:</u>

- 9 lies at $V[4,7]$.

- So,

$V[4,7] = \max\{V[3,7], V[3, 7-5] + 7$

$= \max\{9, V[3,2] + 7\}$

$= \max\{9, 1 + 7\}$

$= \max\{9, 8\}$

- We don't add item 4 as $V[i,w] = V[i-1, w]$ is true.

·) Now, we check item 3 :

·) 9 lies at $V[3,7]$

·) So,

$$V[3,7] = \max\{V[2,7],\ V[2,7-4]+5\}$$
$$= \max\{5,\ 4+5\}$$
$$= \max\{5,\ 9\}$$

·) We include 3rd item, since the condition $V[3,7] = V[2,7]$ is false.

·) Reduce Bag capacity :

$w = 7-4$

$\boxed{w = 3}$

·) ∴ Max Profit (i) - Profit (i)

=) 9 - 5

=) $\boxed{4}$ → we will backtrack to see where 4 lies in the table and at which item.

•) Now, we check for item 2 :

•) 4 lies at $V[2,3]$

•) So,

$$V[2,3] = \max\{ V[1,3],\ V[1, 3-3] + 4 \}$$
$$= \max\{ V[1,3],\ V[1,0] + 4 \}$$
$$= \max\{ 1,\ 0 + 4 \}$$
$$= \max\{ 1, 4 \}.$$

•) we include 2nd item as, condition $V[2,3] = V[1,3]$ is false.

•) Reduce Bag weight :

$W = 3 - 3$

$\boxed{W = 0}$

•) : Max profit (i) - profit (i)

⇒ 4 - 4

⇒ 0

•) we have achieved max profit = 13 with minimum weight by including items :

2, 3, 5.

•) The sequence :

i1	i2	i3	i4	i5
0	1	1	0	1

Ans.

g)

Member 1:- Samad

Member 3:- Umer

Sam, u, m, e

$S = \{19, 1, 13, 21, 13, 5\}$

Sum = 72

target = $\frac{sum}{2} = \frac{72}{2} = 36$; even so partition possible

∵ we need to find elements that adds upto 36

∵ we can't solve it drawing table as there are 36 columns which is not possible to draw at paper, so we will do it directly.

Recurrence Relation

if ($j < S[i-1]$)

$m[i][j] = m[i-1][j]$

else

$m[i][j] = m[i-1][j]$ or $m[j][i-1]$

by looking at array we can see that there is no possible subset with given sum.

(n) Road cutting Problem

Given:

length [] = {1, 2, 3, 4, 5, 6, 7, 8}
Price [] = {1, 5, 8, 9, 10, 16, 18, 20}
Road length = 8 = n

define T [n+1],
initialize T[i] = 0 for i = 0 to n

i \ j	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	1	5	6	10	11	15	16	20
3	0	1	5	8	10	13	16	17	21
4	0	1	5	8	10	13	16	17	21
5	0	1	5	8	10	13	16	17	21
6	0	1	5	8	10	13	16	17	21
7	0	1	5	8	10	13	16	18	21
8	0	1	5	8	10	13	16	18	21

length[i]

$$T[i] = \begin{cases} T[i] & ; j > \text{length}[i] \\ \max(T[i], \text{price}[j-1] + T[i-j]) & ; j <= \text{length}[i] \\ 0 & ; i, j == 0 \end{cases}$$

∴ Maximum profit that we can get is 21, using rods of length 3, 3, 2

Ans

(i)

Given :

$S = \{1, 5, 6, 8\}$

target = 13

define T [target + 1],

initially set T[0] = 0 & T[1 ... target] = ∞

Iterations $\quad i \rightarrow \quad T[i]$

j	S[j]	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	1	0	1	2	3	4	5	6	7	8	9	10	11	12	13
1	5	0	1	2	3	4	1	2	3	4	5	2	3	4	5
2	6	0	1	2	3	4	1	1	2	3	4	5	2	2	3
3	8	0	1	2	3	4	1	1	2	1	2	3	2	2	2

$$T[i] = \begin{cases} \min(T[i], T[i - S[j] + 1] & i - S[j] >= 0 \\ T[i] & i - S[j] < 0 \end{cases}$$

∴ Minimum number of coins = $\boxed{T[\text{target}] = 2 \ (8 \ \& \ 5)}$

Ans.

(j)

dictionary = { i, like, ice, cream, icecream, mobile, apple }

s = ilike apple , n = 10

define dp [n+1]

set dp[0] = true;

Iterations (1 = True & 0 = false)

i \ j	0	1	2	3	4	5	6	7	8	9	10
1	1	1	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0
5	1	1	0	0	0	1	0	0	0	0	0
6	1	1	0	0	0	1	0	0	0	0	0
7	1	1	0	0	0	1	0	0	0	0	0
8	1	1	0	0	0	1	0	0	0	0	0
9	1	1	0	0	0	1	0	0	0	0	0
10	1	1	0	0	0	1	0	0	0	0	1

Final Array:

dp [T, T, F, F, F, T, F, F, F, T }

dp[10] returns true.

String can be segmented.