

G.H. RAISONI COLLEGE OF ENGG, NAGPUR

2020-2021 ODD TERM

CAE-2 EXAMINATION FOR SPLIT - II COURSES

DEPARTMENT: ARTIFICIAL INTELLIGENCE

SEM / SEC : 3 / A DATE: 28/08/2020

SUBJECT: Design & Analysis of Algorithms

ROLLNO: A-58

NAME: SHIVAM TAWARI

REG. NO.: 2019AIE1117028

Q. 2.

a.

Given :

Job Index	1	2	3	4	5
Job	J1	J2	J3	J4	J5
Deadline	2	1	3	2	1
Profit	60	100	20	40	20

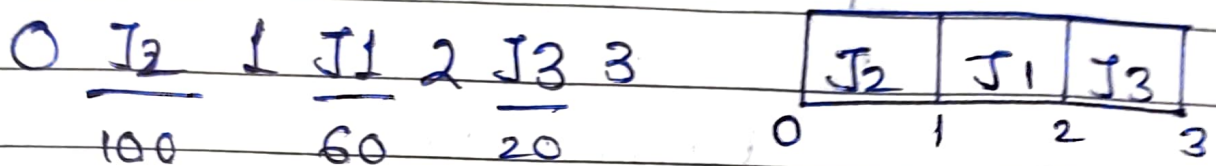
Now we sort the jobs according to their profit in descending order.

*Shivam Tawari*

28/08/2020

Job Index	1	2	3	4	5
Job	J2	J1	J4	J3	J5
Deadline	1	2	2	3	1
Profit	100	60	40	20	20

Maximum deadline value  
 $d_{max} = 3$



Required Job  $\Rightarrow J_2 \rightarrow J_1 \rightarrow J_3$

Max Profit  $\Rightarrow J_2 + J_1 + J_3$

$$\Rightarrow 100 + 60 + 20$$

$$\Rightarrow 180$$

If we look at job J2, it has a deadline 1. This means we have to complete job J2 in time slot 1 if we want to earn its profit.

*Alex*  
*Harris*

28/08/2020

Similarly, if we look at job J1 it has a deadline 2. This means we have to complete job J1 on or before time slot 2 in order to earn its profit.

Similarly, if we look at job J3 it has a deadline 3. This means we have to complete job J3 on or before time slot 3 in order to earn its profit.

Q.1.

a.

$$x = \langle 0, 0, 1, 1, 1, 0, 1 \rangle$$

$$y = \langle 0, 1, 1, 0, 1, 0, 1 \rangle$$



28/08/2020



		X	0	0	1	1	1	0	1
		-1	0	1	2	3	4	5	6
Y	-1	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1
1	1	0	1	2	2	2	2	2	2
1	2	0	1	2	3	3	3	3	3
0	3	0	1	2	2	3	3	4	4
1	4	0	1	2	3	3	4	4	5
0	5	0	1	2	3	3	4	5	5
1	6	0	1	2	3	4	4	5	6

Longest Subsequence: 01101

Length: 6

LCS Algorithm:

Step 1: Start

Step 2: Input string X and Y.

Step 3: Construct a matrix L with  $(m+1, n+1)$  dimensions and  $i=0$ .

Step 4: Check if  $i \leq m$ , if true then set  $j=0$  else goto Step 8.

*Handwritten signature*

28/08/2020

Step 5: if either  $i=0$  or  $j=0$ , set  $L[i][j]=0$  else if  $x[i-1] = y[j-1]$ , then set  $L[i][j] = L[i-1][j-1] + 1$  else set  $L[i][j]$  with max of left and upper cell.

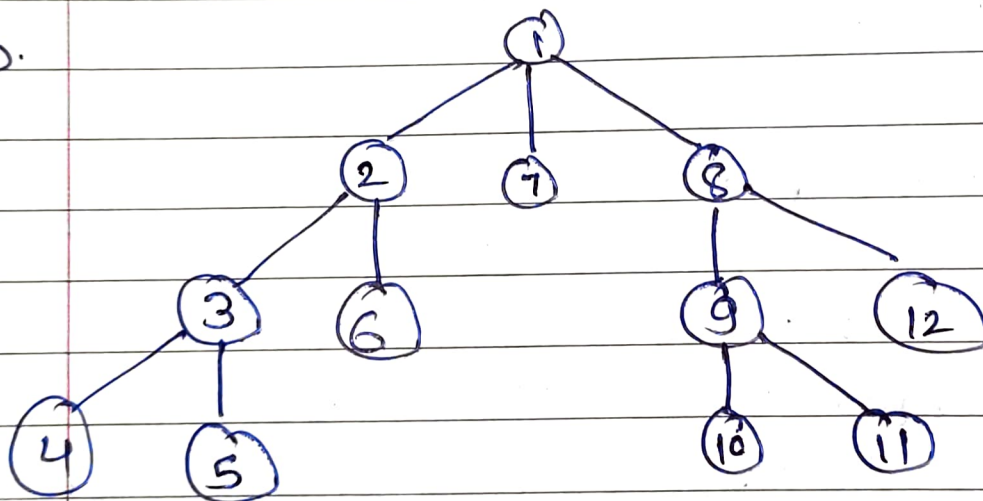
Step 6: Increment  $j$ .

Step 7: If  $j \leq m$ , go to Step 5 else goto Step 4 with  $i++$ .

Step 8: Print maximum possible subsequence length and LCS.

Step 9: Stop.

b.



*Handwritten signature*

28/08/2020

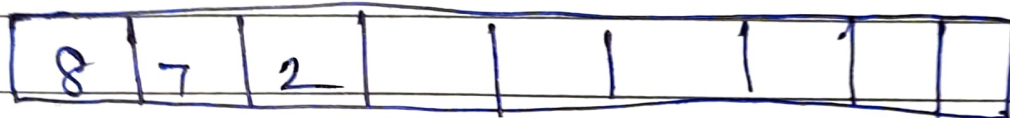
① ~~visited~~ ~~Process~~ ~~Thompson's~~ ~~Process~~

~~Thompson's~~

Stack

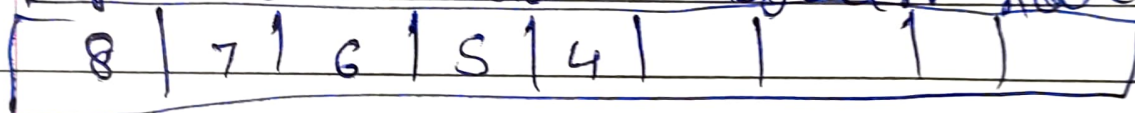


Pop 1 and add 4 adjacent node, add 1 in



result

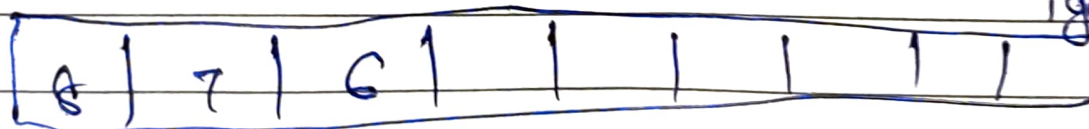
Pop 2 and add 4 adjacent node



Pop 3 and add 4 adjacent node



Pop 5 and add adjacent node, else



ignore

~~After~~

28/08/2020



Pop 6 and add its adjacent nodes

8	7	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Pop 7

8	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Pop 8

12	9	1	1	1	1	1	1
----	---	---	---	---	---	---	---

Pop 9

9	10	11	1	1	1	1
---	----	----	---	---	---	---

Pop 10

12	10	1	1	1	1	1	1
----	----	---	---	---	---	---	---

Pop 11

12	1	1	1	1	1	1	1
----	---	---	---	---	---	---	---

*[Signature]*

28/08/2020

Pop 12



Path for DFS is 1, 2, 3, 4, 5, 6, 7,  
8, 9, 10, 12.

Harsh

28/08/2020