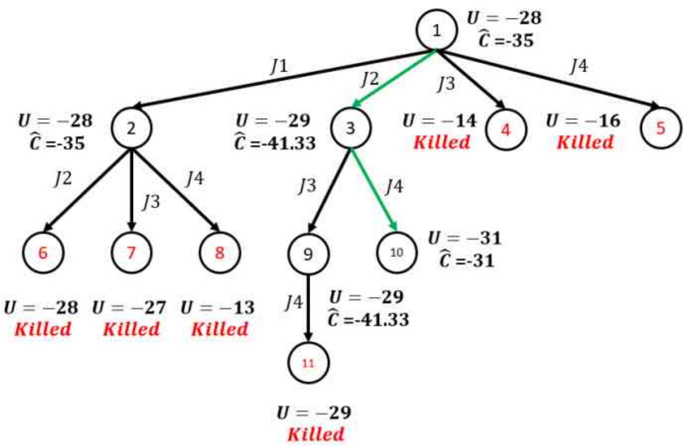
Q1

Solve the Knapsack Problem using FIFOBB, assume knapsack capacity is W =8. Show how queue is used for node creation in the state space tree.

Item	Profit	Weight
1	13	4
2	15	2
3	14	4
4	16	6

Answer

Ans:



Job 1, Job 2, Job 3 are selected with profit 12

Node 1: $U = -(13 + 15) = -28$, Upper = -28

$\hat{C}$ = -(28+(8-(4+2))14/4) =-35 , Node 2, 3, 4, 5 created

Node 2: $U = -(13 + 15) = -28$,

$\hat{C}$ = -(28+(8-(4+2))14/4) =-35 , Node 6,7,8 created

Node 3: $U = -(15 + 14) = -29$, Upper = -29

$\hat{C}$ = -(29+(8-(2+4))16/2) =-41.33 , Node 9,10 created

Node 4: $U = -(14) = -14$, Node killed U> Upper

Node 5: $U = -(16) = -16$, Node killed U> Upper

Node 6: $U = -(13 + 15) = -28$, Node killed U>Upper

$\hat{C}$ = -(28+(8-(4+2))14/4) =-35 , Node 2, 3, 4, 5 created

Node 7: $U = -(13 + 14) = -27$, Node killed U> Upper

Node 8: $U = -(13) = -13$, Node killed U> Upper

$\hat{C}$ = -(29+(8-(2+4))16/2) =-41.33 , Node 9,10 created

Node 9: $U = -(15 + 14) = -29$,

$\hat{C}$ = -(29+(8-(2+4))16/2) =-41.33 , Node 11 created

Node 10: $U = -(15 + 16) = -31$, Upper=-31

$\hat{C}$ = -(31+(8-(2+6))) =-31

Node 11: $U = -(15 + 14) = -29$, Node killed U> Upper

Answer: Job2 amd Job4 are selected with **Maximum profit = 31**

Q2

Find the existence of a pattern P in the given string S (assign digits A-C as 0-2), using Rabin Karp algorithm. For hash function use Mod 13. Find out how many spurious hits does the algorithm encounter in the **Text =ABCBBCABCBAAABCCAACB** when looking for the pattern **Pattern = CCA**?

Answer

Ans

H(X) = {P[2]*10^2+P[1]*10^1+P[0]}mod 13.

Assign value A=0, B=1, C=2

H(P) = (200+20+0)mod 13 =220=12.

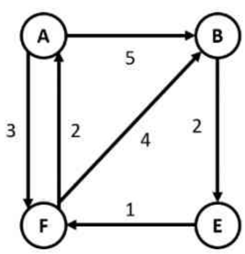
Text: **ABCBBCABCBAAABCCAACB**

ABC	12 MOD 13 =	12	spurious hits
BCB	121 MOD 13 =	4	
CBB	211 MOD 13 =	3	
BBC	112 MOD 13 =	8	
BCA	120 MOD 13 =	3	
CAB	201 MOD 13 =	6	
ABC	12 MOD 13 =	12	spurious hits
BCB	121 MOD 13 =	4	
CBA	210 MOD 13 =	2	
BAA	100 MOD 13 =	9	
AAA	0 MOD 13 =	0	
AAB	1 MOD 13 =	1	
ABC	12 MOD 13 =	12	spurious hits
BCC	122 MOD 13 =	5	
CCA	220 MOD 13 =	12	Hit
CAA	200 MOD 13 =	5	
AAC	2 MOD 13 =	2	
ACB	21 MOD 13 =	8	

Total number of spurious hits is 3 for the patten CCA.

Q3

Consider a logistics manager tasked with optimizing transportation routes for a delivery company that operates in a city with a complex network of roads. Your goal is to minimize the distance for packages to reach their destinations by finding the shortest paths between all pairs of locations. The transportation route is represented as weighted directed graph given below. Find the shortest paths between all pairs of locations, considering the varying distance, which helps company to delivery operation.



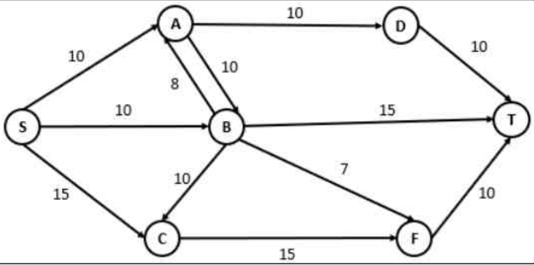
Answer

Base Condition (D0):
0 5 inf 3
inf 0 2 inf
inf inf 0 1
2 4 inf 0

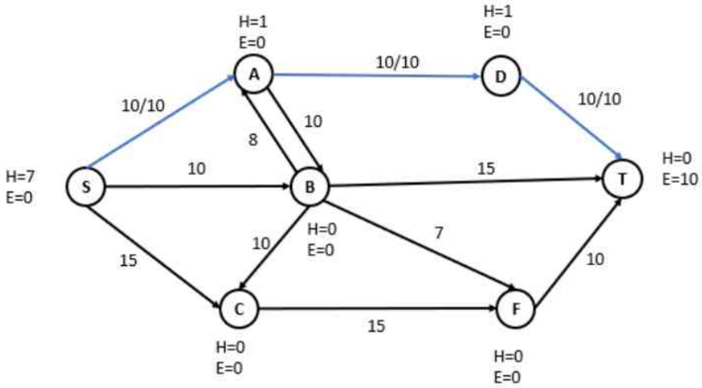
Iteration 1: 0 5 inf 3 inf 0 2 inf inf inf 0 1 2 4 inf 0	Shortest path from 0 to 0 with distance 0: 0 Shortest path from 0 to 1 with distance 5: 0 -> 1 Shortest path from 0 to 3 with distance 3: 0 -> 3 Shortest path from 1 to 1 with distance 0: 1 Shortest path from 1 to 2 with distance 2: 1 -> 2 Shortest path from 2 to 2 with distance 0: 2 Shortest path from 2 to 3 with distance 1: 2 -> 3 Shortest path from 3 to 0 with distance 2: 3 -> 0 Shortest path from 3 to 1 with distance 4: 3 -> 1 Shortest path from 3 to 3 with distance 0: 3
Iteration 2: 0 5 7 3 inf 0 2 inf	Shortest path from 0 to 0 with distance 0: 0 Shortest path from 0 to 1 with distance 5: 0 -> 1 Shortest path from 0 to 2 with distance 7: 0 -> 1 -> 2

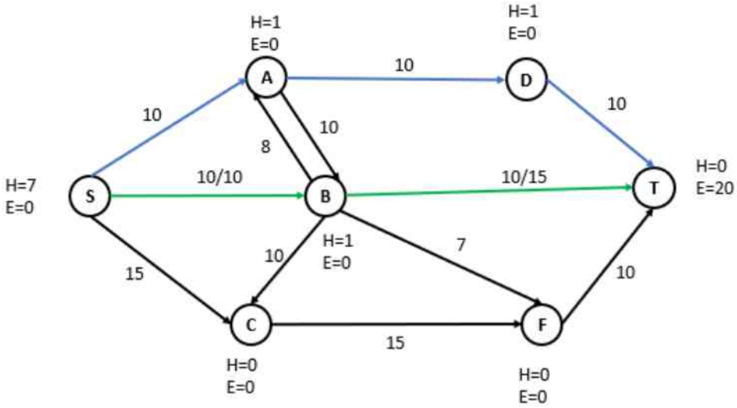
	inf inf 0 1 2 4 6 0	Shortest path from 0 to 3 with distance 3: 0 -> 3 Shortest path from 1 to 1 with distance 0: 1 Shortest path from 1 to 2 with distance 2: 1 -> 2 Shortest path from 2 to 2 with distance 0: 2 Shortest path from 2 to 3 with distance 1: 2 -> 3 Shortest path from 3 to 0 with distance 2: 3 -> 0 Shortest path from 3 to 1 with distance 4: 3 -> 1 Shortest path from 3 to 2 with distance 6: 3 -> 1 -> 2 Shortest path from 3 to 3 with distance 0: 3
	Iteration 3: 0 5 7 3 inf 0 2 3 inf inf 0 1 2 4 6 0	Shortest path from 0 to 0 with distance 0: 0 Shortest path from 0 to 1 with distance 5: 0 -> 1 Shortest path from 0 to 2 with distance 7: 0 -> 1 -> 2 Shortest path from 0 to 3 with distance 3: 0 -> 3 Shortest path from 1 to 1 with distance 0: 1 Shortest path from 1 to 2 with distance 2: 1 -> 2 Shortest path from 1 to 3 with distance 3: 1 -> 2 -> 3 Shortest path from 2 to 2 with distance 0: 2 Shortest path from 2 to 3 with distance 1: 2 -> 3 Shortest path from 3 to 0 with distance 2: 3 -> 0 Shortest path from 3 to 1 with distance 4: 3 -> 1 Shortest path from 3 to 2 with distance 6: 3 -> 1 -> 2 Shortest path from 3 to 3 with distance 0: 3
	Iteration 4: 0 5 7 3 5 0 2 3 3 5 0 1 2 4 6 0	Shortest path from 0 to 0 with distance 0: 0 Shortest path from 0 to 1 with distance 5: 0 -> 1 Shortest path from 0 to 2 with distance 7: 0 -> 1 -> 2 Shortest path from 0 to 3 with distance 3: 0 -> 3 Shortest path from 1 to 0 with distance 5: 1 -> 2 -> 3 -> 0 Shortest path from 1 to 1 with distance 0: 1 Shortest path from 1 to 2 with distance 2: 1 -> 2 Shortest path from 1 to 3 with distance 3: 1 -> 2 -> 3 Shortest path from 2 to 0 with distance 3: 2 -> 3 -> 0 Shortest path from 2 to 1 with distance 5: 2 -> 3 -> 1 Shortest path from 2 to 2 with distance 0: 2 Shortest path from 2 to 3 with distance 1: 2 -> 3 Shortest path from 3 to 0 with distance 2: 3 -> 0 Shortest path from 3 to 1 with distance 4: 3 -> 1 Shortest path from 3 to 2 with distance 6: 3 -> 1 -> 2 Shortest path from 3 to 3 with distance 0: 3

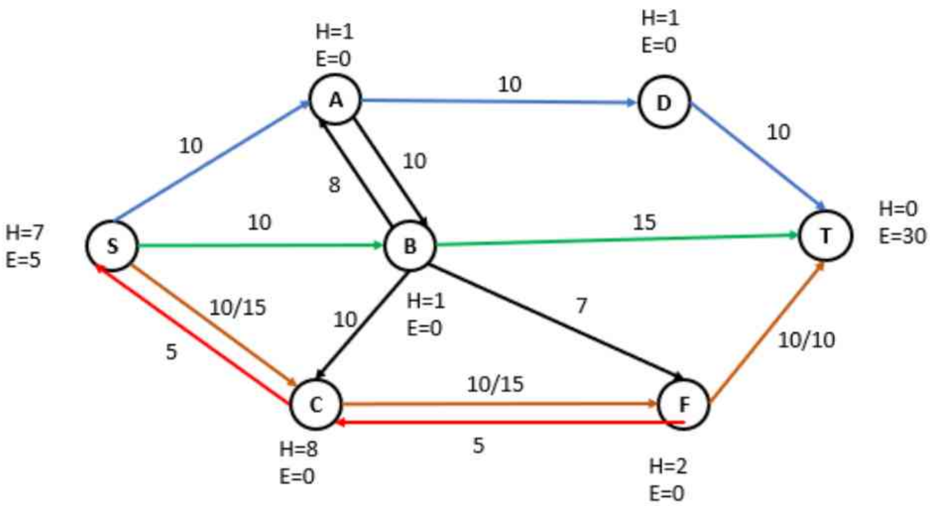
Q4	In water distribution systems, we need to find the maximum amount of water that can be supplied from source S to destination T through a network pipes with capacity limitations. Given a directed graph G=(V,E) representing a water distribution system, where V is set of vertices and E is the set of edges, each edge (u,v) has a capacity c(u,v) representing the maximum water flow that can be supplied through the network pipe. Use Push Relabel algorithm to find the maximum water flow that can be supplied from node S to node T using given graph.

Answer: Ans: Max flow 30







Q5	Find whether the following line segments intersect or not using cross product. a. L1 :{**(1,23) & (10,15)**} and L2 :{**(4,10) & (6,20)**} b. L3 :{**(4,5) & (7,10)**} and L4 :{**(1,1) & (5,5)**} c. L5 :{**(1,1) & (10,10)**} and L6 :{**(3,3) & (5,5)**} d. L7 :{**(1,1) & (10,10)**} and L8 :{**(5,8) & (3,3)**}
Answer	Ans: a. $P11 \times (\overline{P12\ P21})$

$$= \begin{vmatrix} (4-1) & (10-1) \\ (10-23) & (15-23) \end{vmatrix} = \begin{vmatrix} 3 & 9 \\ -13 & -8 \end{vmatrix} = +ve$$

$P11 \times (\overline{P12\,P22})$

$$= \begin{vmatrix} (6-1) & (10-1) \\ (20-23) & (15-23) \end{vmatrix} = \begin{vmatrix} 5 & 9 \\ -3 & -8 \end{vmatrix} = -ve$$

$$L1\ and\ L2\ intersect$$

b. $P31 \times (\overline{P32\,P41})$

$$= \begin{vmatrix} (1-4) & (7-4) \\ (1-5) & (10-5) \end{vmatrix} = \begin{vmatrix} -3 & 3 \\ -4 & 5 \end{vmatrix} = -ve$$

$P31 \times (\overline{P32\,P42})$

$$= \begin{vmatrix} (5-4) & (7-4) \\ (5-5) & (10-5) \end{vmatrix} = \begin{vmatrix} 1 & 3 \\ 0 & 5 \end{vmatrix} = +ve$$

$$L3\ and\ L4\ intersect$$

c. $P51 \times (\overline{P52\,P61})$

$$= \begin{vmatrix} (3-1) & (10-1) \\ (3-1) & (10-1) \end{vmatrix} = \begin{vmatrix} 2 & 9 \\ 2 & 9 \end{vmatrix} = 0$$

$P51 \times (\overline{P52\,P62})$

$$= \begin{vmatrix} (5-1) & (10-1) \\ (5-1) & (10-1) \end{vmatrix} = \begin{vmatrix} 4 & 9 \\ 4 & 9 \end{vmatrix} = 0$$

Bounding Box for L5

$$\widehat{P51} = \{\min(1,10), \min(1,10)\} = \{(1,1)\}$$

$$\widehat{P52} = \{\max(1,10), \max(1,10)\} = \{(10,10)\}$$

Bounding Box for L6

$$\widehat{P61} = \{\min(3,5), \min(3,5)\} = \{(3,3)\}$$

$$\widehat{P62} = \{\max(3,5), \max(3,5)\} = \{(5,5)\}$$

If $\{P51, P52, \widehat{P51}, \widehat{P52}\}$ *and* $\{P61, P62, \widehat{P61}, \widehat{P62}\}$

(10>3) and (5>1) and (10>3) and (5>1) hence *L5 and L6 intersect*

d. $P71 \times (\overline{P72\,P81})$

$$= \begin{vmatrix} (5-1) & (10-1) \\ (8-1) & (10-1) \end{vmatrix} = \begin{vmatrix} 4 & 9 \\ 7 & 9 \end{vmatrix} = -ve$$

$P71 \times (\overline{P72\,P82})$

$$= \begin{vmatrix} (3-1) & (10-1) \\ (3-1) & (10-1) \end{vmatrix} = \begin{vmatrix} 2 & 9 \\ 2 & 9 \end{vmatrix} = 0$$

$$L1\ and\ L2\ intersect$$

```
Line1 and Line2 points:
p1 (x y): 1 23
p2 (x y): 10 15
p3 (x y): 4 10
p4 (x y): 6 20
Position of P3 with respect to P1P2: right
Position of P4 with respect to P1P2: left
Position of P1 with respect to P3P4: left
Position of P2 with respect to P3P4: right
The line segments intersect.
------------------------------------------------
Line3 and Line4 points:
p1 (x y): 4 5
p2 (x y): 7 10
p3 (x y): 1 1
```