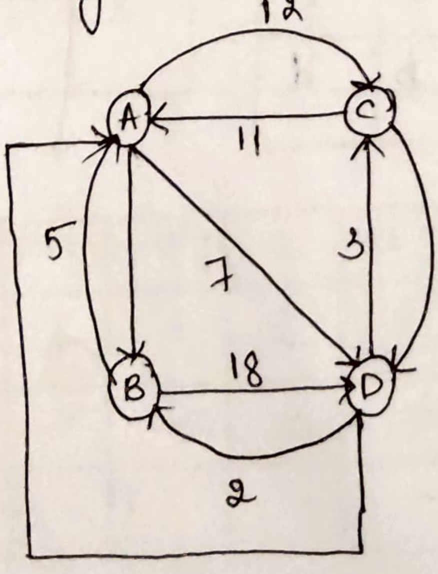
vertex j from source

```
{
  for j: =1 to n do
     s[i]: = false;
  dist[v]: = 0;
  for k: =1 to n do
  {
     choose u such that s[u]=false and dist[u] is the
     minimum;
     s[u]: = true;
     for each adjacent vertex w of u with s[w] =false
                                                    do
     if (dist [w] > dist[u]+cost [u, w]) then
          dist[w] := dist[u]+cost[u, w]
  }
}
```

∴ The time Complexity for Dijkstra's Algorithm is $O((|V|+|E|)\log|V|)$.

9. Solve the given

10.



The Matrix that can be Written from the graph is

$$\begin{array}{c|c} & \begin{matrix} A & B & C & D \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{bmatrix} 0 & 4 & 12 & 7 \\ 5 & 0 & 0 & 18 \\ 11 & 0 & 0 & 6 \\ 10 & 2 & 3 & 0 \end{bmatrix} \end{array}$$

$i_{11} = 0$.

$n = 4$ (4×4 matrices).

$g(i, \phi) = c_{i1}$ (As there are no intermediate nodes).

$g(2, \phi) = c_{21} = 5$

$g(3, \phi) = c_{31} = 11$

$g(4, \phi) = c_{41} = 10$.

Let Size of the set $|s| = 1$ $i \neq 1$; $i \notin s$

So, $i = 2, 3, 4$.

if $i = 2$

i) $g(2, \{3\})$

$g(i, s) = \min\limits_{j \in s} \{c_{ij} + g(j, s - \{j\})\}$.

$= \min\limits_{3 \in s} \{c_{23} + g(3, \phi)\} = \min\{0 + 11\}$

$= 11$.

ii) $g(2, \{4\}) = \min\limits_{4 \in s} \{c_{24} + g(4, \phi)\} = \min\{18 + 10\}$

$= 28$.

iii) if $i = 3$

i) $g(3, \{2\}) = \min\limits_{2 \in s} \{c_{32} + g(2, \phi)\} = \min\{0 + 5\}$

$= 5$.

ii) $g(3, \{4\}) = \min\limits_{4 \in s} \{c_{34} + g(4, \phi)\} = \min\{6 + 10\}$

$= 16$.

if $i = 4$

i) $g(4, \{2\}) = \min\limits_{2 \in s} \{c_{42} + g(2, \phi)\} = \min\{2 + 5\}$

$= 7$.

ii) $g(4, \{3\}) = \min\limits_{3 \in s} \{c_{43} + g(3, \phi)\} = \min\{3 + 11\}$

$= 14$.

let size of the set be |S| : 2 i / 1

i) $g(2, \{3,4\}) = \min\limits_{3,4 \in j} \begin{cases} c_{23} + g(3, \{4\}) = 0 + 16 = 16 \\ c_{24} + g(4, \{3\}) = 18 + 14 = 32 \end{cases}$

$\min(16, 32) = 16.$

at vertex $j = 3$

ii) $g(3, \{2,4\}) = \min\limits_{2,4 \in j} \begin{cases} c_{32} + g(2, \{4\}) = 0 + 28 = 28 \\ c_{34} + g(4, \{2\}) = 6 + 7 = 13 \end{cases}$

$\min(13, 28) = 13.$

iii) $g(4, \{2,3\}) = \min\limits_{2,3 \in j} \begin{cases} c_{42} + g(2, \{3\}) = 2 + 11 = 13 \\ c_{43} + g(3, \{2\}) = 3 + 5 = 8 \end{cases}$

$\min(8, 13) = 8.$

let size of the set be |S| := 3

$g(1, \{2,3,4\}) = \min\limits_{2,3,4 \in j} \begin{cases} c_{12} + g(2, \{3,4\}) = 0 + 16 = 16 \\ c_{13} + g(3, \{2,4\}) = 12 + 13 = 25 \\ c_{14} + g(4, \{2,3\}) = 7 + 8 = 15 \end{cases}$

$= \min \begin{cases} 16 \\ 25 \\ 15 \end{cases}$

$= 15.$

'J' values that minimizes' $g(i, V - \{1\})$.

Algorithm TSP(G, n)
// G is the graph
// n is the no. of vertices
// Cost [1..n, 1..n] is the cost matrix
// tsp [1..n, 1..2ⁿ] is the table of costs of subtours.
// tour [1..n] contains the vertices in the tour.
{
    for i := 1 to n do
        for S := ϕ to {2, . . . . , n} do
            if ( 1∈S && i∈S ) then continue;

```
if (S = φ) then
    g(i, S) := cost[i, 1];
else
{
    g(i, S) := min [c_ij + g[j, S - {j}]]
               j∈S
    next[S] := j;
}
tour[1] := 1; S := {2, ....., n}
for i := 2 to n do
{
    tour[i] := next[S];
    S := S - next[S];
}
}
```

∴ The Time Complexity is $O(n^2 2^n)$.

10. ~~The~~ Solve the given two Sequences using Longest Common Subsequence. X = {1, 0, 0, 1, 0, 1, 0, 1} and Y = {0, 1, 0, 1, 1, 0, 1, 1, 0}.

Given m = 8   n = 9.   $1 \le i \le 8$.

if y is empty

c[1, 0] = 0

c[2, 0] = 0

c[3, 0] = 0

c[4, 0] = 0

c[5, 0] = 0

c[6, 0] = 0

c[7, 0] = 0

c[8, 0] = 0

c[1, 0] = 0.

if x is empty

$1 \le j \le 9$

c[0, 1] = 0

c[0, 2] = 0

c[0, 3] = 0

c[0, 4] = 0

c[0, 5] = 0

c[0, 6] = 0

c[0, 7] = 0

c[0, 8] = 0

c[0, 9] = 0.

We have,

$$c[i,j] = \begin{cases} 0, & \text{if } i=0, j=0. \\ c[i-1,j-1]+1 & \text{if } i,j>0 \ \& \ x_i = y_j \\ \max(c[i-1,j], c[i,j-1]) & \text{if } i,j>0 \ \& \ x_i \neq y_j. \end{cases}$$

$x[1] \neq Y[1] \Rightarrow \max(c[0,1], c[1,0])$
$= 0+0$
$= 0.$

$x[1] = Y[2] \Rightarrow c[0,1]+1 = 1.$

$x[1] \neq Y[3] \Rightarrow \max(c[0,3], c[1,2])$
$= \max(0,1) = 1.$

$x[1] = Y[4] \Rightarrow c[0,3]+1 = 1.$

$x[1] = Y[5] \Rightarrow \max(c[0,5], c[1,4])$
$= \max(0,1) = 1.$

$x[1] \neq Y[6] \Rightarrow \max(c[0,6], c[1,5])$
$= \max(0,1) = 1.$

$x[1] \cancel{\neq} = Y[7] \Rightarrow c[0,6]+1 = 1.$

$x[1] = Y[8] \Rightarrow c[0,7]+1 = 0+1$
$= 1.$

$x[1] \neq Y[9] \Rightarrow \max(c[0,9], c[1,8])$
$= \max(0,1) = 1.$

$x[2] = Y[1] \Rightarrow c[1,0]+1 = 1.$

$x[2] \neq Y[2] \Rightarrow \max(c[1,2], c[2,1]) = \max(1,1) = 1.$

$x[2] = Y[3] \Rightarrow c[1,2]+1 = 1+1 = 2.$

$x[2] \neq Y[4] \Rightarrow \max(c[1,4], c[2,3]) = \max(1,2)$
$= 2.$

$x[2] \neq Y[5] \Rightarrow \max(c[1,5], c[2,4]) = \max(1,2)$
$= 2.$

$x[2] = Y[6] \Rightarrow c[1,5]+1 = 1+1 = 2.$

$x[2] \neq Y[7] \Rightarrow \max(c[1,7], c[2,6]) = \max(1,2)$
$= 2.$

$x[2] \neq Y[8] \Rightarrow \max(c[1,8], c[2,7]) = \max(1,2) = 2.$

$x[2] = Y[9] \Rightarrow c[1,8]+1 = 1+1 = 2.$

$x[3] = Y[1] \rightarrow c[2,0]+1 = 1$

$x[3] \neq Y[2] \rightarrow \max(c[2,2], c[3,1])$
$= \max(1,1) = 1.$

$x[3] = Y[3] \rightarrow c[2,2]+1 = 1+1 = 2.$

$x[3] \neq Y[4] \rightarrow \max(c[2,4], c[3,3])$
$= \max(2,2) = 2.$

$x[3] \neq Y[5] \rightarrow \max(c[2,5], c[3,4])$
$= \max(2,2) = 2.$

$x[3] = Y[6] \rightarrow c[2,5]+1 = 2+1 = 3.$

$x[3] \neq Y[7] \Rightarrow \max(c[2,7], c[3,6])$
$= \max(2,3) = 3.$

$x[3] \neq Y[8] \Rightarrow \max(c[2,8], c[3,7])$
$= \max(2,3) = 3.$

$x[3] = Y[9] \Rightarrow c[2,8]+1 = 2+1 = 3.$

$x[4] \neq Y[1] \Rightarrow \max(c[3,1], c[4,0])$
$= \max(1,0) = 1.$

$x[4] = Y[2] \Rightarrow c[3,1]+1 = 1+1$
$= 2.$

$x[4] \neq Y[3] \Rightarrow \max(c[3,3], c[4,2])$
$= \max(2,2) = 2.$

$x[4] = Y[4] \Rightarrow c[3,3]+1 = 2+1$
$= 3.$

$x[4] = Y[5] \Rightarrow c[3,4]+1 = 2+1$
$= 3.$

$x[4] \neq Y[6] \Rightarrow \max(c[3,6], c[4,5])$
$= \max(3,3) = 3.$

$x[4] = Y[7] \Rightarrow c[3,6]+1 = 3+1$
$= 4.$

$x[4] = Y[8] \Rightarrow c[3,7]+1 = 3+1$
$= 4$

$x[4] \neq Y[9] \Rightarrow \max(c[3,9], c[4,8])$
$= \max(3,4)$
$= 4.$

$x[5] = y[1] \Rightarrow c[4,0]+1 = 0+1 = 1.$

$x[5] \neq y[2] \Rightarrow \max(c[4,2], c[5,1]) = \max(2,1)$
$= 2.$

$x[5] = y[3] \Rightarrow c[4,2]+1 = 2+1 = 3.$

$x[5] \neq y[4] \Rightarrow \max(c[4,4], c[5,3]) = \max(3,3)$
$= 3.$

$x[5] \neq y[5] \Rightarrow \max(c[4,5], c[5,4]) = \max(3,3)$
$= 3.$

$x[5] = y[6] \Rightarrow c[4,5]+1 = 3+1 = 4.$

$x[5] \neq y[7] \Rightarrow \max(c[4,7], C[5,6]) = \max(4,4)$
$= 4.$

$x[5] \neq y[8] \Rightarrow \max(c[4,7], c[5,7]) = \max(4,4)$
$= 4.$

$x[5] = y[9] \Rightarrow c[4,8]+1 = 4+1 = 5.$

$x[6] \neq y[1] \Rightarrow \max(c[5,1], c[6,0]) = \max(1,0) = 1.$

$x[6] = y[2] \Rightarrow c[5,1]+1 = 1+1 = 2.$

$x[6] \neq y[3] \Rightarrow \max(c[5,3], c[6,2]) = \max(3,2) = 3.$

$x[6] = y[4] \Rightarrow c[5,3]+1 = 3+1 = 4.$

$x[6] = y[5] \Rightarrow c[5,4]+1 = 3+1 = 4.$

$x[6] \neq y[6] \Rightarrow \max(c[5,6], c[6,5]) = \max(4,4) = 4.$

$x[6] = y[7] \Rightarrow c[5,6]+1 = 4+1 = 5.$

$x[6] = y[8] \Rightarrow c[5,7]+1 = 4+1 = 5$

$x[6] \neq y[9] \Rightarrow \max(c[5,9], c[6,8]) = \max(5,5) = 5.$

$x[7] = y[1] \Rightarrow c[6,0]+1 = 1.$

$x[7] \neq y[2] \Rightarrow \max(c[\cancel{0}[6,2], c[7,1]) = \max(2,1) = 2.$

$x[7] = y[3] \Rightarrow c[6,2]+1 = 2+1 = 3.$

$x[7] \neq y[4] \Rightarrow \max(c[6,4], c[7,3]) = \max(4,3) = 4.$

$x[7] \neq y[5] \Rightarrow \max(c[6,5], c[7,4]) = \max(4,4) = 4.$

$x[7] = y[6] \Rightarrow c[6,5]+1 = 4+1 = 5.$

$x[7] \neq y[7] \Rightarrow \max(c[6,7], c[7,6]) = \max(5,5) = 5.$

$x[7] \neq y[8$

$x[7] = y[9$

$x[8] \neq y[1$

$x[8] = y[2$

$x[8] \neq y[$

$x[8] = y[$

$x[8] = y[$

$x[8] \neq y[$

$x[8] = y[$

$x[8] = y[$

$x[8] \neq y$

| | |
|---|---|
| 1 | 0 |
| 0 | 0 |
| 0 | 0 |
| 1 | 0 |
| 0 | 0 |
| 1 | 0 |
| 0 | |
| 1 | |

$x[7] \neq y[8] \Rightarrow \max(c[6,8], c[7,7]) = \max(5,5) = 5.$

$x[7] = y[9] \Rightarrow c[6,8]+1 = 5+1 = 6.$

$x[8] \neq y[1] \Rightarrow \max(c[7,1], c[8,0]) = \max(1,0) = 1.$

$x[8] = y[2] \Rightarrow c[7,1] + 1 = 1+1 = 2.$

$x[8] \neq y[3] \Rightarrow \max(c[7,3], c[8,2]) = \max(3,2) = 3.$

$x[8] = y[4] \Rightarrow c[7,3]+1 = 3+1 = 4$

$x[8] = y[5] \Rightarrow c[7,4]+1 = 4+1 = 5.$

$x[8] \neq y[6] \Rightarrow \max(c[7,6], c[8,5]) = \max(5,5) = 5.$

$x[8] = y[7] \Rightarrow c[7,6]+1 = 5+1 = 6.$

$x[8] = y[8] \Rightarrow c[7,7]+1 = 5+1 = 6.$

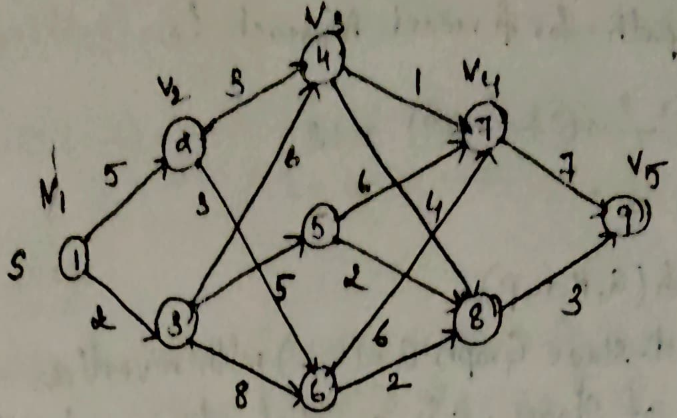
$x[8] \neq y[9] \Rightarrow \max(c[7,9], c[8,8]) = \max(6,6) = 6.$

| | | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 0 | 0 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 3 | 3 |
| 1 | 0 | 1 | 2 | 2 | 3 | 3 | 3 | 4 | 4 | 4 |
| 0 | 0 | 1 | 2 | 3 | 3 | 3 | 4 | 4 | 4 | 5 |
| 1 | 0 | 1 | 2 | 3 | 4 | 4 | 4 | 5 | 5 | 5 |
| 0 | 0 | 1 | 2 | 3 | 4 | 4 | 5 | 5 | 5 | 6 |
| 1 | 0 | 1 | 2 | 3 | 4 | 5 | 5 | 6 | 6 | 6 |

100110

LCS

```
Algorithm LCS(x[1:m], y[1:n])
{
    c[0:m, 0:n]
    for i:=0 to m do
    {
        c[i,0]:=0; b[i,0]:=SKIPY; //initialize Column 0.
    }
    for j:=0 to n do
    {
        c[0,j]:=0; b[0,j]:=SKIPY; //initialize row 0.
    }
    for i:=1 to m do
    {
        for j:=1 to n do
        {
            if (x[i]=y[j]) (characters are matched)
            {
                c[i,j]:=c[i-1,j-1]+1; //take x[i] and y[i] for LCS
                b[i,j]:=ADDXY;
            }
            else if(c[i-1,j] ≥ c[i,j-1]) //x[i] not in LCS
            {
                c[i,j]:>=c[i-1,j];
                b[i,j]:=SKIP x;
            }
            else //Y[j] not in LCS
            {
                c[i,j]:=c[i,j-1];
                b[i,j]:=SKIP y;
            }
        }
    }
    return c[m,n];
}
```




Forward

4th stage

$\text{Cost}(4,7) = c(7,9) = 7$
$\text{Cost}(4,8) = c(8,9) = 3$

3rd stage

$$\text{cost}(3,4) = \min_{7,8 \in V_4} \begin{cases} c(4,7) + \text{Cost}(4,7) = 1+7 = 8 \\ c(4,8) + \text{cost}(4,8) = 3+4 = 7 \end{cases} \quad 7$$

$$\text{cost}(3,5) = \min_{7,8 \in V_4} \begin{cases} c(5,7) + \text{Cost}(4,7) = 6+7 = 13 \\ c(5,8) + \text{Cost}(4,8) = 2+3 = 5 \end{cases} \quad 5$$

$$\text{Cost}(3,6) = \min_{7,8 \in V_4} \begin{cases} c(6,7) + \text{cost}(4,7) = 6+7 = 13 \\ c(6,8) + \text{cost}(4,8) = 2+3 = 5 \end{cases} \quad 5$$

2nd stage:

$$\text{Cost}(2,2) = \min_{4,6 \in V_3} \begin{cases} c(2,4) + \text{Cost}(3,4) = 3+7 = 10 \\ c(2,6) + \text{Cost}(3,6) = 3+5 = 8 \end{cases} \quad 8!$$

$$\text{cost}(2,3) = \min_{4,5,6 \in V_3} \begin{cases} c(3,4) + \text{Cost}(3,4) = 6+7 = 13 \\ c(3,5) + \text{Cost}(3,5) = 5+5 = 10 \\ c(3,6) + \text{Cost}(3,6) = 8+5 = 13 \end{cases} \quad 10.$$

1st stage:

$$\text{cost}(1,1) = \min_{2,3 \in V_2} \begin{cases} c(1,2) + \text{Cost}(2,2) = 5+8 = 13 \\ c(1,3) + \text{Cost}(2,3) = 2+10 = 12 \end{cases} \quad 12$$

$\rightarrow$ $d(1,1) \rightarrow l$ that minimizes the cost $(1,1)$ is 3.

$d(2,3) = 5$

$d(3,5) = 5$

~~d~~

Minimum cost path for forward Approach is,

$1 \xrightarrow{2} 3 \xrightarrow{5} 5 \xrightarrow{2} 8 \xrightarrow{3} 9 = 12$

Algorithm:

Algorithm FGraph(G, k, n, p).
// The Input is a k-Stage Graph G = (V, E) with n vertices indexed in order of Stages. E is a Set of edges and c[i,j] is the Cost of <i,j>. P[1:k] is a minimum cost path.

{

Cost[n] : = 0.0;

for j: = n-1 to Step-1 do

{ // Compute cost [j]

Let r be the vertex Such that <j, r> is an edge of G and

c[j, r] + Cost[r] is minimum.

Cost[j]: = c[j, r] + Cost[r];

d[j] : = r;

}

// find a minimum Cost path.

P[1]: =1 P[k]: =n;

for j: =2 to k-1 do P[j]: =d[P[j-1]];

}

Backward

2nd stage:

bcost(2,2) = c(1,2) = 5

bcost (2,3) = c(1,3) = 2

3rd stage:

$$b\,cost(3,4) = \min_{l \in 2,3} \begin{cases} bcost(2,3) + c(2,4) = 5+3 = 8 \\ bcost(2,4) + c(3,4) = 2+6 = 8 \end{cases}$$

$$bcost(3,5) = \min_{l \in 3} \left\{ bcost(2,3) + c(3,5) = 2+5 = 7. \right.$$

$$bcost(3,6) = \min_{l \in 2,3} \left\{ \begin{array}{l} bcost(2,2) + c(2,6) = 5+3 = 8 \\ bcost(2,3) + c(3,6) = 2+8 = 10 \end{array} \right\} 8$$

## 4th stage:

$$bcost(4,7) = \min_{l \in 4,5,6} \left\{ \begin{array}{l} bcost(3,4) + c(4,7) = 8+1 = 9 \\ bcost(3,5) + c(5,7) = 7+6 = 13 \\ bcost(3,6) + c(6,7) = 8+6 = 14 \end{array} \right\} 9$$

$$bcost(4,8) = \min_{l \in 4,5,6} \left\{ \begin{array}{l} bcost(3,4) + c(4,8) = 8+4 = 12 \\ bcost(3,5) + c(5,8) = 7+2 = 9 \\ bcost(3,6) + c(6,8) = 8+2 = 10 \end{array} \right\} 9$$

## 5th stage:

$$bcost(5,9) = \min_{l \in 7,8} \left\{ \begin{array}{l} bcost(4,7) + c(7,9) = 9+7 = 16 \\ bcost(4,8) + c(8,9) = 9+3 = 12 \end{array} \right\} 12$$

$d(5,9) \rightarrow l$ that minimize cost$(5,9)$ is 8

$d(4,8) = 5$

$d(3,5) = 3$

$d(2,3) = 1.$

Minimum Cost path for Backward Approach is

$$1 \leftarrow 3 \leftarrow 5 \leftarrow 8 \leftarrow 9$$

## Algorithm:

```
Algorithm BGraph(G,K,n,p)
{
  bcost[ ]:=0.0;
  for j:=2 to n do
  { //Compute bcost[j]
    let r be vertex Such that <r,j> is an edge of 'G'
    bcost[r]+c[r,j] is minimum.
    bcost[j]:=bcost[r]+c[r,j];
    d[j]:=r;
  }
  finds a minimum Cost path.
  P[1]:=1    P[K]:=n;
} for j:=K-1 to 2 do P[j]:=d[P[j+1]];
```