**//FloyedWarshall**

**/\***

**Complexity: O(n^3)**

**\*/**

void FloydWarshall(int n)

{

for(int k=1;k<=n;k++)

for(int i=1;i<=n;i++)

for(int j=1;j<=n;j++)

cost[i][j]=min(cost[i][j], (cost[i][k]+cost[k][j]));

}

**// FloydWarshallFlow**

**/\***

**Return all possible source-destination maximum flow using only one path.**

**If there is no connection from u to v, then cost[u][v] should be initialized to zero.**

**\*/**

void FloydWarshallFlow(int n)

{

for(int k=1;k<=n;k++)

for(int i=1;i<=n;i++)

for(int j=1;j<=n;j++)

cost[i][j]=max(cost[i][j], min(cost[i][k], cost[k][j]));

}