**Program 11**

**Implement Warshall’s algorithm using dynamic programming**

**Code:**

#include<stdio.h>

#include<conio.h>

#include<math.h>

int max(int,int);

void warshal(int p[10][10],int n)

{

int i,j,k;

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

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

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

p[i][j]=max(p[i][j],p[i][k]&&p[k][j]);

}

int max(int a,int b)

{

;

if(a>b)

return(a); else

return(b);

}

void main()

{

int p[10][10]= {999},n,e,u,v,i,j;

printf("\n Enter the number of vertices:");

scanf("%d",&n);

printf("\n Enter the number of edges:");

scanf("%d",&e);

for (i=1;i<=e;i++)

{

printf("\n Enter the starting and ending vertex of the edge %d:",i);

scanf("%d%d",&u,&v);

p[u][v]=1;

}

printf("\n Matrix of input data: \n");

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

{

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

printf("%d\t",p[i][j]);

printf("\n");

}

warshal(p,n);

printf("\n Transitive closure: \n");

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

{

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

printf("%d\t",p[i][j]);

printf("\n");

}

getch();

}

**Graph:**

**Adjacency matrix: Transitive closure:**

0 0 1 0 0 0 1 0

1 0 0 1 1 1 1 1

0 0 0 0 0 0 0 0

0 1 0 0 1 1 1 1

2→1 , 1→3 2→3

2→4 , 4→2 2→2

4→2 , 2→1 4→11

4→2 , 2→1 , 1→3 4→3

4→2 , 2→4 4→4

**Output:**

