**PROG 3-A**

#include

<stdio.h>

#include<limits.h>

#define V 4

voidfloydWarshall(intgraph[V][V])

{

intdist[V][V];

for(inti=0;i<V;i++)for(int j=0; j<V;j++) dist[i][j]

= graph[i][j];

for(int k=0;k<V;k++) for (inti=0;i<V; i++)

for(intj=0;j<V; j++)

if(dist[i][k]!=INT\_MAX&&dist[k][j]!=INT\_MAX&& dist[i][k]+dist[k][j]

<dist[i][j])dist[i][j]=dist[i][k]+ dist[k][j]

("Shortestdistancesbetweeneverypairofvertices:\n"); for

(int i = 0; i < V; i++)

{

for(intj=0; j<V; j++)

{

if(dist[i][j]==INT\_MAX)

printf("INF\t");

else

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

}

printf("\n");

}

}

intmain(){

int graph[V][V]={{0,INT\_MAX,3,INT\_MAX},

{2,0,INT\_MAX,INT\_MAX},

{INT\_MAX,7,0,1},{6,INT\_MAX,INT\_MAX,0}};

floydWarshall(graph);

return 0;

}

**PROG 3-B**

#include<stdio.h>

intn,a[10][10],p[10][10];

voidpath()

{

int i,j,k;

for(i=0;i<n;i++)

for(j=0;j<n;j++)

p[i][j]=a[i][j];

for(k=0;k<n;k+

+)

for(i=0;i<n;i++)

for(j=0;j<n;j++)

if(p[i][k]==1&&p[k][j]==1)

p[i][j]=1;

}

voidmain()

{

inti,j;

printf("Enterthenumberofnodes:");

scanf("%d",&n);printf("\nEntertheadjacencymatrix:\n");for(i=0;i<n;i++)

for(j=0;j<n;j++)

scanf("%d",&a[i][j]);path();

printf("\nThepathmatrixisshownbelow\n");

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

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

printf("\n");

}

}