NAME :- Mithil Raj

USN :- 1BM19CS086.

ADA :- Lab test 1 (Batch 3).

4th program :-

Write a program to do the following :-

(a) print all the nodes reachable from a given starting node in a digraph. using BFS method.

```
#include <stdio.h>
  void bfs (int j;
int a[10][10] ; vis[10], n;
void main () {
  int i, j, srt;
printf( " enter the number of vertices \n");
Scanf ("%d", &n)
printf(" enter the adjacent matrix \n");
for (; i=1 ; ; i>n ; i++){
for( j=1 ; j<=n ; j+1 ) {
Scanf( "%d", &a[i][j]) ;
  }
 }
printf ("enter the source node :\t");
Scanf( "%d" ; &src);
printf(" nodes reacable from %d vertices
is ", src);
```

```
bfs(src);
getch();
}
void bfs(int v){
int q[10], f=0, r=0, u, i, j;
vis[v]=1;
q[r]=v;
while(f<=r){
u=q[f];
printf("%d\t", u);
for(i=1; i<=n; i++){
if(a[u][i]==1 && vis[i]==0){
vis[i]=1;
r=r+1;
q[r]=i;
}
}
f=f+1;
}
}
```

(b) check wheather a given graph is connected or not using DFS:

```
#include<stdio.h>
#include<conio.h>
int a[20][20], vis[20], n;
void dfs(int v)
{
int i;
vis[v]=1;
```

```
for(i=1;i<=n;i++)
if(a[v][i]&&!vis[i])
{
dfs(i);
}
}
Void main()
{
int i,j,count=0;
printf("\n enter the number of vertices:
  ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
vis[i]=0;
for(j=1;j<=n;j++)
a[i][j]=0;
}
printf("\n enter the ~~number~~ adjacent
  matrix\n");
for(i=1;i<=n;i++)
for(j=1;j<=n;j++)
scanf("%d",&a[i][j])
dfs(1);
printf("\n");
for(i=1;i<=n;i++)
{
if(vis[i])
count++;
}
if(count==n)
printf("\n the graph is connected\n");
```

else
printf("\nth graph is not connected\n");
getch();
}.

(b) ——— X ———

for graph connected :-

				4 (vertices)
0	1	0	1	
1	0	1	0	
0	1	0	1	
1	0	1	0	

for graph disconnected. 5 (vertices)

:-

0	1	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	9	9
0	0	0	0	0	0
0	0	0	0	0	0

(a)

0	1	0	1	
1	0	1	0	
0	1	0	1	
1	0	1	0	