Name: Mufid Vahora

Roll no. : 20bce307

Course code and name: 2cs302 Data Structures and Algorithm

Practical no. : 8

Code:

```
#include<stdio.h>
#include <stdlib.h>
struct Node
{
 int data;
 struct Node *next;

}*front=NULL,*rear=NULL;
void enqueue(int x)
{
 struct Node *t;
 t=(struct Node*)malloc(sizeof(struct Node));
if(t==NULL)
 printf("Queue is FUll\n");
 else
 {
 t->data=x;
 t->next=NULL;
 if(front==NULL)
 front=rear=t;
 else
 {
 rear->next=t;
```

```
 rear=t;
 }
 }

}
int dequeue()
{
 int x=-1;
 struct Node* t;

 if(front==NULL)
 printf("Queue is Empty\n");
else
 {
 x=front->data;
 t=front;
 front=front->next;
 free(t);
 }
 return x;
}
int isEmpty()
{
 return front==NULL;
}

void BFS(int A[][8],int start,int n){
```

```
int i=start;
 int vis[8]={0};
 printf("%d,",i);
 vis[i]=1;
 enqueue(i);

 while(!isEmpty())
 {
 i=dequeue();
 for(int j=1;j<n;j++){
 if(A[i][j]==1 && vis[j]==0){
printf("%d,",j);
 vis[j]=1;
 enqueue(j);
 }
 }
 }
 printf("\n");
}
int main()
{
 int A[8][8] = {{0, 0, 0, 0, 0, 0, 0, 0},
{0, 0, 1, 1, 1, 0, 0, 0},  {0, 1, 0, 1, 0,
0, 0, 0},  {0, 1, 1, 0, 1, 1, 0, 0},  {0, 1,
0, 1, 0, 1, 0, 0},  {0, 0, 0, 1, 1, 0, 1,
1},  {0, 0, 0, 0, 0, 1, 0, 0},  {0, 0, 0, 0,
0, 1, 0, 0}};
 for(int i=1;i<8;i++){
```

```
BFS(A,i,8);  }

return 0; }
```

Output: