```
#include<stdio.h>
#include<stdlib.h>

void  readGraph(int n,int edg,int a[][n])
{
      int i,j;
      for(i=0;i<n;i++)
      {
          for(j=0;j<n;j++)
          {
               a[i][j]=0;
          }
      }
      for(int k=1;k<=edg;k++)
      {
             printf("Enter the V(i,j) of edge-%d :",k);
             scanf("%d%d",&i,&j);
             a[i][j]=a[j][i]=1;
      }
}
void printGraph(int n,int a[][n])
{
      for(int i=0;i<n;i++)
      {
          for(int j=0;j<n;j++)
          {
               printf("%3d",a[i][j]);
          }
          printf("\n");
      }
}
void enqueue(int q[],int s,int *F,int*R,int n)
{
      if(*R>n)
      {
        printf("Queue is Overflow\n");
      }
      else
      {
             (*R)++;
             q[*R]=s;
      }
}
int dequeue(int q[],int *F,int *R)
{

      if(*F>*R)
      {
          printf("Queue is underflow\n");
      }
      else
      {

         (*F)++;
         return q[*F];
      }
}
void breadthFirstScrh(int n,int s,int a[][n],int b[])
 {
```

```
      int *visit=(int*)calloc(n,sizeof(int));
      int q[n],F=-1,R=-1;
      visit[s]=1;

    enqueue(q,s,&F,&R,n);
    int i=0,k;

    if(n==0)
    {
        return;
    }
      while (F<=R)
      {
        k=dequeue(q,&F,&R);
        b[i]=k;
        i++;

          for(int j=0;j<n;j++)
          {
                  if(a[k][j]==1 && visit[j]==0)
                  {
                visit[j]=1;
              enqueue(q,j,&F,&R,n);
              }
          }
      }

 }
void printBFsTrav(int b[],int n)
{
     if(n==0)
    {
        printf("EMPTY GRAPH\n");
        return;
    }
     for(int i=0;i<n;i++)
       {
              printf("%3d",b[i]);
       }
}
int main()
{
     int n,e;
     printf("Enter the no.of vertices:");
     scanf("%d",&n);

     int a[n][n],s;
     printf("Enter the number of edges: ");
     scanf("%d",&e);
     readGraph(n,e,a);


     printf("Graph in AdjMatrix:\n");
     printGraph(n,a);


    source: printf("Enter the starting node: ");
                  scanf("%d",&s);
```

```
            int b[n];
            breadthFirstScrh(n,s,a,b);

            printf("GRAPH Traversal BFS:\n");
            printBFsTrav(b,n);

                  printf("\nEnter opt:\n1.repeat\n2.exit\n<=");
            scanf("%d",&e);
                  if(e==1)
                  {
                        goto source;
                  }
}
```