Q1. DFS and BFS using adjacency matrix

```
Aim: C program for DFS and BFS algorithm #include<stdio.h>
```

```
int q[20],top=-1,front=-1,rear=-1,a[20][20],vis[20],stack[20];
int delete();
void add(int item);
void bfs(int s,int n);
void dfs(int s,int n);
void push(int item);
int pop();
void main()
int n,i,s,ch,j;
char c,dummy;
printf("ENTER THE NUMBER VERTICES");
scanf("%d",&n);
for(i=1;i \le n;i++)
for(j=1;j \le n;j++)
printf("ENTER 1 IF %d HAS A NODE WITH %d ELSE 0 ",i,j);
scanf("%d",&a[i][j]);
printf("ADJACENCY MATRIX IS\n");
for(i=1;i \le n;i++)
for(j=1;j \le n;j++)
printf(" %d",a[i][j]);
printf("\n");
do
for(i=1;i \le n;i++)
vis[i]=0;
printf("\nMENU");
printf("\n1.B.F.S");
printf("\n2.D.F.S");
printf("\nENTER YOUR CHOICE");
scanf("%d",&ch);
printf("ENTER THE SOURCE VERTEX :");
scanf("%d",&s);
switch(ch)
case 1:bfs(s,n);
break;
```

```
case 2:
dfs(s,n);
break;
printf("DO U WANT TO CONTINUE(Y/N)?");
scanf("%c",&dummy);
scanf("%c",&c);
}while((c=='y')||(c=='Y'));
}
void bfs(int s,int n)
int p,i;
add(s);
vis[s]=1;
p=delete();
if(p!=0)
printf(" %d",p);
while(p!=0)
for(i=1;i \le n;i++)
if((a[p][i]!=0)&&(vis[i]==0))
add(i);
vis[i]=1;
p=delete();
if(p!=0)
printf(" %d ",p);
for(i=1;i \le n;i++)
if(vis[i]==0)
bfs(i,n);
}
void add(int item)
if(rear=19)
printf("QUEUE FULL");
else
if(rear = -1)
q[++rear]=item;
front++;
q[++rear]=item;
int delete()
```

```
int k;
if((front>rear)||(front==-1))
return(0);
else
k=q[front++];
return(k);
}
void dfs(int s,int n)
int i,k;
push(s);
vis[s]=1;
k=pop();
if(k!=0)
printf(" %d ",k);
while(k!=0)
for(i=1;i \le n;i++)
if((a[k][i]!=0)&&(vis[i]==0))
push(i);
vis[i]=1;
k=pop();
if(k!=0)
printf(" %d ",k);
for(i=1;i \le n;i++)
if(vis[i]==0)
dfs(i,n);
void push(int item)
if(top==19)
printf("Stack overflow ");
else
stack[++top]=item;
int pop()
int k;
if(top=-1)
return(0);
else
k=stack[top--];
return(k);
}
}
```

RESULT:

C:\Users\STUDENT\Desktop\ada2\bin\Debug\ada2.exe

```
ENTER THE NUMBER VERTICES 4
V1 1 with V2 1 0/10
V1 1 with V2 2 0/11
V1 1 with V2 3 0/11
V1 1 with V2 4 0/11
V1 2 with V2 1 0/10
V1 2 with V2 2 0/10
V1 2 with V2 3 0/10
V1 2 with V2 4 0/11
V1 3 with V2 1 0/10
V1 3 with V2 2 0/10
V1 3 with V2 3 0/10
V1 3 with V2 4 0/10
V1 4 with V2 1 0/10
V1 4 with V2 2 0/10
V1 4 with V2 3 0/11
V1 4 with V2 4 0/10
ADJACENCY MATRIX IS
0 1 1 1
0001
0000
 0010
MENU
1.B.F.S
2.D.F.S
ENTER YOUR CHOICE1
ENTER THE SOURCE VERTEX :1
1 2 3 4 DO U WANT TO CONTINUE(Y/N) ? y
MENU
1.B.F.S
2.D.F.S
ENTER YOUR CHOICE2
ENTER THE SOURCE VERTEX :1
1 4 3 2 DO U WANT TO CONTINUE(Y/N) ? _
```

```
15/6/23 Exp x
 DFS & BFS using adjacency meatoix
sol"
#include < stdio. h >
IN also, top = -1, foot = -1, oca = -1, a[20][20], vis[20],
       Stack[20];
it delete();
wid add ( it item);
void bes (ints, int n);
void off (its, it is);
void push (no item);
(1) gog tu
roid mah ()
 ind n,s, i, ch, j;
dial c. di
point ("Exte the number of vertice");
scart (" 1/d", ln)!
Ασ (121; icn; i++).
 808 (j=1; j <=n; j++) {
 postf("enta 1 of "had has a node with "had else 0",
       (, i);
 scand ( " 1.d , &a[i][i]);
point ("The Adjacency Matolala");
10x(1=1; ic=n; it+)5
4x()=1; i<=n; i+t) &
point (" /d", a [i][j]); }
```

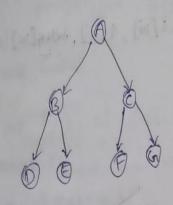
```
port (" \n");
do
 for(i=1; ic=n; i++)
 vis[i]=0;
 print (" \n Menu");
 (".B. P.S") " Horag
  printf("/n2.DFS);
  Potet ("Enta govo choice");
  scand ("/.d", och);
   popul ("Fitel the source vestex");
   scant ("1.d", 85);
   switch (cn)
   case 1 ; bfs (s, n);
   borak;
   case 21 olfs (s, n);
    break)
 )(" ( (M/n) surteres of trace up of ) throat
  Scarf ("·1·c", od);
  sout ( · · [· ( , &c) )
   while (((== 2 )/11 (== 2/N'))),
 void bis (hts, htn)
   ing th
   add (s);
```

```
Vis [5]21;
po delete ();
A (P1=0)
pointf ("/·d", p);
 while (PI=0)
 los (1=1; i <= n; i++)
 A ((a[p][i] 1 = 0) 88 (vir[i] = = 0))
 and(1);
 11=[i] Er
 p= delete();
 if (p1=0)
 printf( " of.d", p);
λος (i=1; i c=n; i++)
 rt (vis[i]==0)
  bfs (i, n);
void acid (ht item)
# (real = = 19)
printf. Dowe of full');
ete
 A (real = = -)
q[++ real]= item;
front ++;
9/[++seal] = item;
```

```
() state (1)
 int k;
# ((food 2 deal) | food = = 1)
 return (0);
else
k= 9 [food ++];
 oution(k);
(n til , 2 til) 240 blor
 ind i,k;
 push (s);
 VY [5] 21;
 k= pop();
 f(k!=0)
 South ("1.01, k);
 while (k! 20)
 for (1=1) (<=n; i++)
 H((a[K][i] ! 20) 80 (NI[i]=20))
  push(i);
  T=[1] sin
 K= pop();
 # (k)=0)
poput ( "1.d ", E);
```

```
Los (121; ican; it+)
 H ( NA [] = > 0)
 olfs (i, n);
 noid push (ht item)
 17 (top==19)
 point ("stack overflow");
 else
 stade[++top]= item;
  int pop()
  int ki
 if (top==-)
  return (0),
  else
  Icestack [top=-];
  oction (k);
Enta number of valities
                            Given Adjacency natoria
VI with VI 1/0 ! 0
V1
         V3
M
      with Ve
               1010
V2
           V2
Ve
                   10
           V3
 V2
           Vu
```

 V_1 V_1 V_2 V_3 V_4 V_5 V_2 V_2 V_4 V_5 V_2 V_2 V_4 V_5 V_6 V_2 V_7 V_7



BFS 1- ABCDERG VI V2 V3 V4 V5 V6 V1 DFS 1- ABDECPG V1 V2 V4 V5 V3 V6 V-

Sp. 1, 3