**3.Write a C program depth first search (DFS) using array.**

```
#include<stdio.h>

void DFS(int);
int G[10][10],visited[10],n;

void main()
{
   int i,j;
   printf("Enter number of vertices:");

         scanf("%d",&n);


         printf("\nEnter adjecency matrix of the graph:");

         for(i=0;i<n;i++)
      for(j=0;j<n;j++)
                        scanf("%d",&G[i][j]);


  for(i=0;i<n;i++)
       visited[i]=0;

   DFS(0);
}

void DFS(int i)
{
   int j;
         printf("\n%d",i);
   visited[i]=1;

         for(j=0;j<n;j++)
      if(!visited[j]&&G[i][j]==1)
         DFS(j);
}
```

**OUTPUT**

```
Enter adjecency matrix of the graph:0 1 1
1 0 1
1 1 0
```

0
1
2

**4.Write a C program breath first search (BFS) using array.**

```
#include<stdio.h>
int a[20][20],q[20],visited[20],n,i,j,f=0,r=-1;
void bfs(int v) {
        for (i=1;i<=n;i++)
          if(a[v][i] && !visited[i])
           q[++r]=i;
        if(f<=r) {
                visited[q[f]]=1;
                bfs(q[f++]);
        }
}
void main() {
        int v;
        printf("\n Enter the number of vertices:");
        scanf("%d",&n);
        for (i=1;i<=n;i++) {
                q[i]=0;
                visited[i]=0;
        }
        printf("\n Enter graph data in matrix form:\n");
        for (i=1;i<=n;i++)
          for (j=1;j<=n;j++)
           scanf("%d",&a[i][j]);
        printf("\n Enter the starting vertex:");
        scanf("%d",&v);
        bfs(v);
        printf("\n The node which are reachable are:\n");
        for (i=1;i<=n;i++)
          if(visited[i])
           printf("%d\t",i); else
           printf("\n Bfs is not possible");
        getch();
}
```

**Output:**

Enter the number of vertices:4

Enter graph data in matrix form:
1 1 1 1
0 1 0 0
0 0 1 0
0 0 0 1

Enter the starting vertex:1

The node which are reachable are:
1
2
3
4