

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

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

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

```
void DFS(int);
```

```
int C[10][10],visited[10]={0},n;
```

```
void main()
```

```
{
```

```
    int h,a;
```

```
    printf("Enter number of vertices: ");
```

```
    scanf("%d",&n);
```

```
    printf("\n Enter the adjacent matrix of the Graph : ");
```

```
    for(h=0;h<n;h++)
```

```
    {
```

```
        for(a=0;a<n;a++)
```

```
            scanf("%d",&C[h][a]);
```

```
    }
```

```
    DFS(0);
```

```
}
```

```
void DFS(int h)
```

```
{
```

```
    int a;
```

```
    printf("\n%d",h);
```

```
    visited[h]=1;
```

```
    for(a=0;a<n;a++)
```

```
        if(!visited[a]&&C[h][a]==1)
```

```
            DFS(a);
```

```
}
```

---

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

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

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

```
int C[20][20],h[20]={0},n,visited[20]={0},a,d,f=0,r=-1;
```

```
void BFS(int m)
```

```
{
```

```
    for(a=0;a<n;a++)
```

```
        if(C[m][a]&&visited[a]==0)
```

```
            h[++r]=a;
```

```
            if(f<=r)
```

```
            {
```

```
                visited[h[f]]=1;
```

```
                BFS(h[f++]);
```

```
            }
```

```
}
```

```
void main()
```

```
{
```

```
    int m;
```

```
    printf("Enter the number of vertices : ");
```

```
    scanf("%d",&n);
```

```
    printf("\nEnter the data of graph in form of matrix:\n ");
```

```
    for(a=0;a<n;a++)
```

```
    {
```

```
        for(d=0;d<n;d++)
```

```
            scanf("%d",&C[a][d]);
```

```
    }
```

```
    printf("\nEnter the start vertex: ");
```

```
    scanf("%d",&m);
```

```
    BFS(m);
```

```
    printf("\n nodes which can be reached are : ");
```

```
    for(a=0;a<n;a++)
```

```
{  
    if(visited[a])  
        printf("%d\t",a);  
    else{  
        printf("cant reach all nodes.BFS is not possible");  
        break;  
    }  
}  
}
```