

## **//Write a program to traverse a graph using BFS method.**

```
#include <stdio.h>

int queue[50], front = -1, rear = -1;
int visited[50];
int graph[50][50];
int n; // number of vertices

void bfs(int start)
{
    int i;
    queue[++rear] = start;
    visited[start] = 1;

    printf("BFS Traversal: ");

    while (front != rear)
    {
        start = queue[++front];
        printf("%d ", start);

        for (i = 0; i < n; i++)
        {
            if (graph[start][i] == 1 && visited[i] == 0)
            {
                queue[++rear] = i;
                visited[i] = 1;
            }
        }
    }
}
```

```
    }  
    }  
}  
}
```

```
int main()
```

```
{
```

```
    int i, j, start;
```

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

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

```
    printf("Enter adjacency matrix:\n");
```

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

```
    {
```

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

```
        {
```

```
            scanf("%d", &graph[i][j]);
```

```
        }
```

```
    }
```

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

```
        visited[i] = 0;
```

```
    printf("Enter starting vertex: ");
```

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

```
    bfs(start);
```

```
    return 0;  
}
```

## OUTPUT:-

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  
  
PS C:\Users\Admin\Documents\1WN24CS181> cd "c:\Users\Admin\Documents\1WN24CS181\" ; if ($?) { gcc BFS.C -o BFS } ; if ($?) { .\BFS }  
Enter number of vertices: 4  
Enter adjacency matrix:  
0 1 1 0  
1 0 0 1  
1 0 0 1  
0 1 1 0  
Enter starting vertex: 0  
BFS Traversal: 0 1 2 3  
PS C:\Users\Admin\Documents\1WN24CS181> 
```