

```

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

#define MAX 10 // Maximum number of buildings (nodes)

int graph[MAX][MAX]; // Adjacency matrix
int visited[MAX];    // Visited array
int queue[MAX];      // Queue for BFS
int front = -1, rear = -1;

void addEdge(int u, int v) {
    graph[u][v] = 1;
    graph[v][u] = 1; // Undirected graph
}

void BFS(int start, int n) {
    int i;
    front = rear = 0;
    queue[rear] = start;
    visited[start] = 1;

    printf("\nBFS Traversal (Campus Path): ");

    while (front <= rear) {
        int current = queue[front++];
        printf("%d ", current);

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

int main()
{
    int n, e, u, v, start, i, j;

    printf("Enter number of buildings (nodes): ");
    scanf("%d", &n);

    printf("Enter number of connections (edges): ");
    scanf("%d", &e);

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

    printf("Enter connections (building1 building2):\n");
    for (i = 0; i < e; i++) {
        scanf("%d%d", &u, &v);
    }
}

```

```
addEdge(u, v);  
}
```

```
printf("Enter starting building number: ");  
scanf("%d", &start);
```

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

```
BFS(start,n);  
return 0;  
}
```