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

#define MAX_VERTICES 100

struct Graph {
    int matrix[MAX_VERTICES][MAX_VERTICES];
    int visited[MAX_VERTICES];
    int numVertices;
};

struct Graph* createGraph(int numVertices) {
    struct Graph* graph = (struct Graph*)malloc(sizeof(struct Graph));
    graph->numVertices = numVertices;
    int i, j;
    for (i = 0; i < numVertices; i++) {
        for (j = 0; j < numVertices; j++) {
            graph->matrix[i][j] = 0;
        }
        graph->visited[i] = 0;
    }
    return graph;
}

void addEdge(struct Graph* graph, int src, int dest) {
    graph->matrix[src][dest] = 1;
    graph->matrix[dest][src] = 1;
}

void DFS(struct Graph* graph, int vertex) {
    printf("%d ", vertex);
    graph->visited[vertex] = 1;
    int i;
    for (i = 0; i < graph->numVertices; i++) {
        if (graph->matrix[vertex][i] && !graph->visited[i]) {
            DFS(graph, i);
        }
    }
}

void main() {
    int numVertices = 5;
    struct Graph* graph = createGraph(numVertices);
```

```
    addEdge(graph, 0, 1);
    addEdge(graph, 0, 2);
    addEdge(graph, 1, 2);
    addEdge(graph, 2, 3);
    addEdge(graph, 1, 3);

    printf("Depth First Traversal starting from vertex 0: ");
    DFS(graph, 0);
}
```

```
Depth First Traversal starting from vertex 0: 0 1 2 3
```