#include <stdio.h>

#include <stdlib.h>

#define MAX 10

struct Graph {

int vertices;

int adj[MAX][MAX];

};

void addEdge(struct Graph\* graph, int src, int dest) {

graph->adj[src][dest] = 1;

graph->adj[dest][src] = 1;

}

void BFS(struct Graph\* graph, int start) {

int visited[MAX] = {0};

int queue[MAX];

int front = 0, rear = 0;

visited[start] = 1;

queue[rear++] = start;

while (front < rear) {

int node = queue[front++];

printf("%d ", node);

for (int i = 0; i < graph->vertices; i++) {

if (graph->adj[node][i] == 1 && !visited[i]) {

visited[i] = 1;

queue[rear++] = i;

}

}

}

}

void DFS(struct Graph\* graph, int node, int visited[MAX]) {

visited[node] = 1;

printf("%d ", node);

for (int i = 0; i < graph->vertices; i++) {

if (graph->adj[node][i] == 1 && !visited[i]) {

DFS(graph, i, visited);

}

}

}

int main() {

struct Graph graph;

graph.vertices = 5;

for (int i = 0; i < MAX; i++) {

for (int j = 0; j < MAX; j++) {

graph.adj[i][j] = 0;

}

}

addEdge(&graph, 0, 1);

addEdge(&graph, 0, 2);

addEdge(&graph, 1, 3);

addEdge(&graph, 1, 4);

int choice, visited[MAX];

while (1) {

printf("\n1. BFS traversal\n2. DFS traversal\n3. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("BFS starting from vertex 0: ");

BFS(&graph, 0);

printf("\n");

break;

case 2:

printf("DFS starting from vertex 0: ");

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

DFS(&graph, 0, visited);

printf("\n");

break;

case 3:

exit(0);

default:

printf("Invalid choice!\n");

}

}

return 0;

}

1. BFS traversal

2. DFS traversal

3. Exit

Enter your choice: 1

BFS starting from vertex 0: 0 1 2 3 4

1. BFS traversal

2. DFS traversal

3. Exit

Enter your choice: 2

DFS starting from vertex 0: 0 1 3 4 2

1. BFS traversal

2. DFS traversal

3. Exit

Enter your choice: 3