40 Hamilton

#include <stdio.h>

#define MAX\_N 100 // Adjust this value as needed

// Function to check if a vertex can be included in the current path

bool isSafe(int graph[][MAX\_N], int n, int path[], int v) {

// Check if the vertex is already included in the path

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

if (path[i] == v) {

return false;

}

}

// Check if there's an edge from the previous vertex to the current vertex

return graph[path[n - 1]][v];

}

// Recursive function to find a Hamiltonian circuit

bool findHamiltonianCircuit(int graph[][MAX\_N], int n, int path[], int pos) {

// Base case: If all vertices are included in the path

if (pos == n) {

// Check if there's an edge from the last vertex to the first vertex for a complete circuit

if (graph[path[n - 1]][path[0]]) {

return true;

} else {

return false;

}

}

// Try all possible vertices

for (int v = 0; v < n; v++) {

// Check if the vertex is safe

if (isSafe(graph, n, path, v)) {

path[pos] = v; // Add the vertex to the path

// Recursively try to find a circuit from the next vertex

if (findHamiltonianCircuit(graph, n, path, pos + 1)) {

return true; // Circuit found

}

// Backtrack if the current path doesn't lead to a circuit

path[pos] = -1; // Remove the vertex from the path

}

}

// No circuit found

return false;

}

// Function to print the Hamiltonian circuit

void printHamiltonianCircuit(int path[], int n) {

printf("Hamiltonian Circuit: ");

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

printf("%d ", path[i]);

}

printf("%d\n", path[0]); // Print the starting vertex again to complete the circuit

}

int main() {

/\*

Example graph (modify as needed):

1

/ \

2-----3

| |

4-----5

\*/

int graph[][MAX\_N] = {

{0, 1, 1, 1, 0},

{1, 0, 1, 0, 1},

{1, 1, 0, 1, 1},

{1, 0, 1, 0, 1},

{0, 1, 1, 1, 0}

};

int n = sizeof(graph[0]) / sizeof(graph[0][0]); // Number of vertices

int path[n];

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

path[i] = -1; // Initialize path as empty

}

if (findHamiltonianCircuit(graph, n, path, 0)) {

printHamiltonianCircuit(path, n);

} else {

printf("No Hamiltonian Circuit Found\n");

}

return 0;

}