# California State University, Fresno

# DEPARTMENT OF COMPUTER SCIENCE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class: | **Algorithms & Data Structures** | | | Semester: | **Spring 2022** |
|  | | | | | |
| Points |  | Document author: | **MingkuanPang** | | |
|  | Author’s email: | **Yafking20 @mail.fresnostate.edu. email** | | |
| Laboratory number: | **10 – BFS\_DFS** | | |
|  | | | | | |

**1. Statement of Objectives**

This lab asks for implementation of BFS and DFS of the graph with maximum size of 10 nodes, numbered from 1 to 10. The input will be the number of nodes and its adjacency list.

**2. Experimental Procedure**

**dfs ():**

Text

Description automatically generated

Procedure dfs will travel through the node list, and if the current node has not been visited yet, then pass it into ads\_ultities with its adjacency list.

**dfs\_ultities():**

Text

Description automatically generated

For dfs\_ultities(), it checks if the passed node is visited, if it has been visited yet, then pass it into visit\_node() procedure, if the passed node has adjacencies, it will keep visiting its adjacencies until the passed node does not have adjacency.

**bfs ():**

Text

Description automatically generated

For bfs procedure, it will pass the nodes in the node list into a first in first out queue. The node comes in first, will be visited first. After the node be visited, its adjacencies will be pushed into queue waiting for be visited.

**visit\_node():**

Text

Description automatically generated

For visit\_node procedure, it will check if the passed node is visited, if it has not been visited yet, then print the node on the screen and mark it as visited.

=**ake\_v()**

Text

Description automatically generated

For make\_v prcudure, it takes an integer “size” as parameter, then it will return an array containing the number from 1 to size. For example, if size is 6, then it will return {1,2,3,4,5,6}. This procedure is made for making the list of nodes.

**m\_stoi ()**

A screenshot of a computer

Description automatically generated with medium confidence

For M\_stoi, it transfers a string into numbers and stores it in a vector.

**3. Analysis**

**Main function**

Text

Description automatically generated

The main function asks user to input the size of the node at the first, then it will ask user to input the adjacencies of the node from the first node to the last node in the node list, enter ‘n’ if the node does not have adjacency. Then it will travel the nodes according to depth first order and breadth first order. After finish travelling it will ask user if the user wants to test the program again (input ‘n’ to quit the program).

**Output**

Text

Description automatically generated

**4. Encountered Problems**

When implement the dfs procedure, if I put the node linked itself (put itself into its adjacency list). The program will enter infinite loop. So, in the dfs\_ultities, I add the code to check if the node is visited before putting it into the recursion loop. If the node has already been visited, then it is not necessary to put it into dfs\_ultities again.

**5. Conclusions**

From this experiment, I went over how to travel the graph in depth first order and breadth first order. I can solve many searching problems such as finding the solution of 8-puzzle with those two algorithms. This lab will help me a lot if I learn AI or some other knowledge requires the understanding of searching problem in the future.

**6. References**

I did not use any reference in this lab.