# California State University, Fresno

# DEPARTMENT OF COMPUTER SCIENCE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class: | **Algorithms & Data Structures** | | | Semester: | **Fall 2021** |
|  | | | | | |
| Points |  | Document author: | **Ojas Deshmukh** | | |
|  | Author’s email: | **ojas@mail.fresnostate.edu email** | | |
| Laboratory number: | **Lab 11** | | |
|  | | | | | |

**1. Statement of Objectives**

We will be implementing a BFS and DFS of graph.

**2. Experimental Procedure**

In this we start traversing from a node to other nodes like a path starting from a fixed path. In BFS we will traverse though the nodes in one level before moving in the next level because the distance between the other level nodes is more compared to the nodes in the same level or closest level. After we set one root or node to start at then we travel tom the other two nodes, and these are then stored under the visited label.

Then we go on to the untraversed nodes that follow the previous nodes and then store them in visited. And as we traverse the algorithms calculates the minimum distances so on.

For BFS, single vertex is chosen at one instance and stored to visited and the next vertex is stored in the list and then visited.

For DFS, is a method based on edge. Stack is the data structure that is used in DFS to perform two steps of algorithms- we will first push the vertex that are visited into the stack and then we will pop the vertices that have been visited if there is no vertex present further.

**3. Analysis**

The time complexity for BFS is of the order O(E+V) where E is number of edges and V is the number of nodes for our algorithm.

The time complexity for DFS is of the order O(E+V) where E is number of edges and V is the number of nodes for our algorithm.

A screenshot of a computer

Description automatically generated

**4. Encountered Problems**

The node implementation was a little difficult. I had to continuously keep track of all the variables.

**5. Conclusions**

The time complexity for BFS is of the order O(E+V) where E is number of edges and V is the number of nodes for our algorithm.

The time complexity for DFS is of the order O(E+V) where E is number of edges and V is the number of nodes for our algorithm.

But for general case the BFS is slower than DFS.

**6. References**

List the references used in this report.