Programmer Manual

A. Problem Description

This program demonstrates a Graph ADT using an Adjacency List representation for the graph. It includes a templated Digraph class with data attributes and member functions implemented according to the specifications in the "graph_specs.txt" file. The program supports both directed and undirected graphs and allows users to perform operations on the graph using a menu-driven interface.

B. Datatypes and Classes

The program uses two categories of data types: uses string as the vertex name data type and int as the edge weight data type. The following subsections address the data types used.

1. Data Members:

- num_of_verticies ● graph_array

2. Members Function:

- Digraph and ~Digraph ● set_size
- getDigraph
- get_size

- isVertex
- isUniEdge
- isBiDirEdge ● addVertex
- deleteVertex

- addUniEdge
- addBiDirEdge
- deleteUniEdge ● deleteBiDirEdge ● printDigraph
- breadth
- depth
- getOneVertex
- getTwoVerticies

C. High Level Program Solution

1. The program begins by creating a Digraph object with two template parameters, V and W, representing the vertex name and weight respectively.
2. The program then greets the user and displays a menu of options to choose from.
3. The user is prompted to select an option by entering the corresponding number.
4. The options include:
    - Set size of the graph (number of vertices)
    - Read the graph from a file
    - Check if a vertex exists in the graph
    - Check if an edge exists in the graph (directed or undirected)
    - Add a vertex to the graph
    - Delete a vertex from the graph (and all its incident edges)
    - Add an edge to the graph (directed or undirected)
    - Delete an edge from the graph (directed or undirected)
    - Print the graph

- Traverse the graph using either Breadth-First Search (BFS) or Depth-First Search (DFS) algorithms starting from a given vertex.
- Exit the program

5. The program then prompts the user to enter the required parameters for the selected option and performs the corresponding action.
6. The program continues to display the menu until the user selects the option to exit.
7. When the program exits, any dynamically allocated memory is freed and the program terminates.

D. Limitations and Suggestions

No limitations.