**Name:** Krishna Sreenivas

**Student** **ID**: 800984436

**Programming** **language** **used:** Python 3.6.0

The project consists of a single file, containing the implementation of the heap and graph related functionalities.

**Description:**

1. The Heap class consists of attributes such as heap list and functions such as bubble up, bubble down, minimum child, push, popping min and constructing heap to carry out the heap related functionalities.
2. The vertex class consists of attributes such as vertex name, adjacency list, visited status and parent information and functions such as adding neighbors, adding vertex, setting parent node etc and others. The graph class consists of functionalities such as adding vertex, adding edge, deleting edge, vertex up/down, edge up/down, printing graph, calculating reachability and calculating shortest path using Dijkstra’s etc and other functionalities.
3. The output of print\_graph, reachable and shortest path using Dijkstra’s functionalities are written to a file.
4. The time complexity of the reachable code would be O(|V|^2(|V|+|E|)). Since the code sorts the outer vertices and then sorts inner vertices and calculates the reachability by calling the appropriate function which then loops over the adjacency list.

**Execution command:**

Enter the below command in command prompt.

*python project\_2.py <graph file> <queries file> <output file>*

**Files:**

Project\_2.py

**Description of Data Structure used:**

1. Custom heap implementation for implementing priority queue. The program also makes use of dictionaries and lists to hold graph and vertex related information.

**Issues:**

1. Dijkstra’s cannot operate on a forest containing 2 or more graphs.
2. When making a vertex down or up, the status of in and out edges (i.e. edge up or edge down) of the corresponding vertex are not set.