**COP 5536 Spring 2015 Programming Project April 15,2015**

**Programming Language:** JAVA (v 1.7)

**IDE:** Eclipse

**Compiler:** javac 1.8.0\_20

**Compilation Instructions:**

1. Extract the .zip file

2. In the terminal, navigate to the directory where the extraction is done

3. Run the command: make clean

4. Run the command: make

For Part 1:

5. Run the command: java ssp <inputFileName> <sourceNode> <destinationNode>

For Part 2:

6. Run the command: java routing <inputGraphFile> <inputIPFile> <sourceNode> <destinationNode>

**Function Prototypes:**

The project includes following classes:

* **ssp.java:** Initiates the part 1 of the project

Input: inputFile, sourceNode, destinationNode

* **routing.java:** Initiates the part 2 of the project

Input: inputGraphFile, inputIPFile sourceNode, destinationNode

* **FibonacciHeap.java:** This class represents the Fibonacci heap used by Dijkstra's SSSP algorithm
* **FibHeapNode.java:** This class represents the Fibonacci heap node used by Fibonacci Heap
* **BinaryTrieNode.java:** This class represents the Binary Trie node used by Binary Trie
* **BinaryTrie.java:** This class represents the Binary Trie used by routing.java
* **Graph.java:** This class generates an undirected graph used by Dijkstra and Dijkstra2

Input: Vertex count and edges

Output: Generates an undirected graph

* **BundleFromDijk.java:** This class represents bundles result from Dijkstra2, used by routing.java
* **Dijkstra.java:** This class implements the Dijkstra’s Single Source Shortest Path algorithm and makes use of the FibonacciHeap and FibHeapNode
* **Dijkstra2.java:** This class implements the Dijkstra’s Single Source Shortest Path algorithm and makes use of the FibonacciHeap and FibHeapNode and return the output as an instance of BundleFromDijk

Class Diagram:

