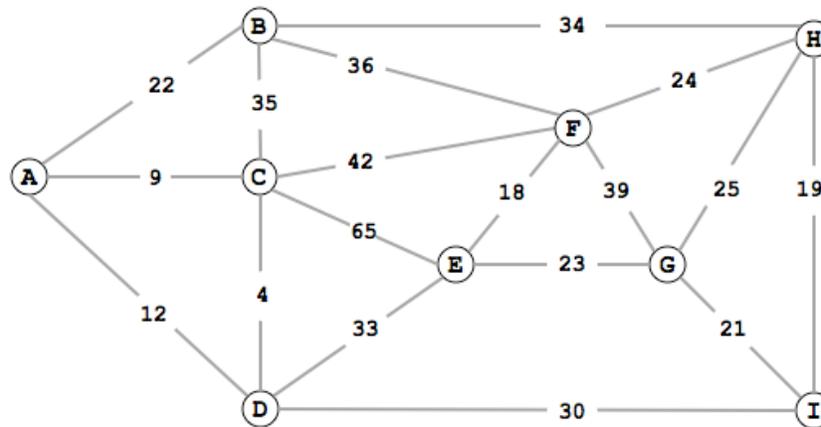


COT5405: Programming Assignment 1 (Spring 2015)

For all programming questions, you should generate the corresponding code in any one of the following language (c, c++, Java, Python) that can be executed under the Eustis2 Unix machine. How to log in to this Unix machine in our Computer Science Division is introduced on 2/12 lecture and slides. For code execution result, you should show the *screenshot image* of your SSH client window containing the code printout in your report.

Submission: (1). A project report; (2). The three programs for Question 2 to 4.

Consider the following undirected graph with weight values on each edge.



1. **(No programming question)** Write down the adjacency matrix representation and link list representation of this graph (in the similar format as the examples of Page 8,9 on lecture slides 'graph.ppt'). The edge value should be shown in those two representation formats, too.
2. **(Shortest Path):** Program a code using Dijkstra's Algorithm to generate the shortest path results starting from node A to all other nodes. Your code should print out the shortest path route and the path value for each destination node. For example, if the shortest path from node A to node E is A -> C -> E, then your code should print out result for this destination node as:
Node E: path value = 74, path is: A -> C -> E
3. **(Minimum Spanning Tree):** Program a code using Kruskal's Algorithm to generate one MST for this graph. Suppose the MST contains three nodes (A,C,E) and the two edges between them, then your code should print out result like:
Minimum Spanning Tree: Total weights on MST edges = 74
Node Set = {A, C, E}, Edge Set = { A-C, C-E}
4. **(Heap and Heapsort):** Considering all the edge values in above graph as an unsorted list. Program a code to heapsort this list of values. Use the array data structure for the heap (as shown in Page 20 in lecture slides 'heapsort.ppt'). Your code should print out this array after the initial "heapify the array" operation (as the array example on Page 20 of slides), and the final sorted result list. In addition, in your report, you should draw the heap binary tree (as the tree example on Page 20 of slides) according to the initial heapify result.