**Directions:** Start early! Programming takes time

**Link state routing 20 points** Write a program (in C/C++) for computing a routing table based on a topology database. This assignment is a simplified version of what a link state router does. Your input will consist of an LSP database. Nodes are denoted by single lower case characters (e.g. 'f', 'k'). Each line of input represents an LSP. The line starts with the node originating the LSP. It is followed by all the neighbors, together with the cost to reach them. Costs are multiple digit decimal numbers. The separator is one space. For example the line “b g 68 a 1 d 26” represents an LSP originated by node b that says that node b can reach node g for a cost of 68 node a for a cost of 1 and node d for a cost of 26 (and that it has no other neighbors. Assume links are unidirectional. Your output is the routing table of the first node (the node that originates the first LSP you receive), one entry per line. Each line has two node names: one destination and the next hop. You must have one line for each node that has an LSP except the first one (to which the routing table you produce belongs). Your code must handle the case when the network is partitioned (i.e. there is no way to get from some nodes to others). When a node is not reachable, the routing table should have zero ('0') as a next hop for that node. You can assume that all nodes have at least one neighbor and that each node mentioned in an LSP has an LSP of its own in the input.

Grading: 0 points if you hand in nothing, 1 point if you hand in something that doesn’t compile (no matter how small the syntax error), 5 points if you hand in something that tries to do link state routing and it compiles. We will test you programs both on the sample inputs on the class web page and on other inputs. We will also look at the source code, but only if it compiles.