For this assignment, you will be writing a small C++ program to implement a static-timing analysis tool using DFS. To simplify your code, you do not need to handle level-sensitive latches or latches; you need merely propagate timing from primary inputs through gates.

To avoid reinventing the wheel, you should start with our network reader. It is documented in network_reader.docx. The file sta.cxx also has a few routines (for creating a stack and using it to find false paths) that will make your life easier. When you're done, turn in just the file sta.cxx via the Provide interface.

You should use two networks:

1. The same small network as in the paper-and-pencil homework (from the file HW2_network1.gmf). Note that this is the same as the file HW1_network1.gmf, except that the PIs now have arrival times (for STA) rather than a sequence of logic values (for simulation). Feel free to compare results with people analyzing this network manually.
2. A small network with a false path (in HW2_mux_false_path.gmf). This is mostly just two muxes, along with a false-path statement.

Your code should run STA and then print the latest arrival time for all nodes.