Important notes:

1. Based on your assignment (online or F2F) you can use the ports in the testbench and instantiate them in your gcn.sv
2. You will need to write the gcn.sv from scratch. The sample gcn.sv was just a template you are responsible for adding all the ports you need in your design. The ports that you use should be an exact match with that of the testbench.
3. Comment out dump_vpd() if it throws any warnings.
4. give_outputs() task is your friend. You can use it to give various inputs to the testbench and see how the outputs of the testbench change. This is so that you can model your inputs in your gcn.sv to give it in accordance with the testbench.
5. give_outputs() **must be commented out** when you are simulating the testbench with your design.
6. For online students remember to uncomment **compare_online();** this compares the
    a. **outputs generated and written** in the memory by gcn.sv against the golden outputs.
7. For F2F students remember to uncomment **compare_F2F();** this compares the **outputs generated and written** in the memory by gcn.sv against the golden outputs.
8. Don't forget to fix the parameters in the params.vh before running your design.
9. You are responsible for generating the addresses to access the memories in the testbench. All be it reading from or writing to. There are ports in the testbench for that.
10. Hint: One way of generating address is through a counter.
11. input_re must be high to read any memory from testbench.
12. input_we must be high to write to the memory in the tb.
13. The matrix given to you is an undirected graph. It is bidirectional. We have provided you with an adj.txt so that you can visualize the graph and also use it as a reference if you are decoding the COO matrix.
14. The port declaration in the main lab 4 pdf document is only a reference tempelate to give you an idea. You should use the port list in the testbench for your design.
15. There are counters (count and countb) that count the number of negative and positive clock cycles in the testbench. You can use this to perform all cycle related calculations in your design.
16. The reset signal is active high. Meaning reset happens when the rst_n goes low and resumes normal operation when rst_n goes high. Design your dffs accordingly.