Practice - 2

1. Simulate the dff.v (it contains the testbench too).

- Important: Once you run ./a.out, it will keep running infinitely, because it is in an always block. You need to hit Ctrl +Z to stop it, else, the vcd will become a large file and will never end.
- Note that in the testbench, the clock and d input are being toggled using the following statements. See the comments in the code
  - always
  - #3 clk=~clk;
  - always
  - #5 d=~d;
- Add a synchronous reset input to this D-Flip Flop. You will need to change the testbench to simulate it
- Create another Verilog module to make this an asynchronous reset. Note the difference between the two types of reset

Practice - 2

2. Simulate the counter.v (it contains the testbench too)

- Important: Once you run ./a.out, it will keep running infinitely, because it is in an always block. You need to hit Ctrl +Z to stop it, else, the vcd will become a large file and will never end.
- This is a 4 bit up-counter.
- Note that, the counter wraps around. Once it reaches “1111”, it goes back to 0000--> you should be able to see why.
- Convert this to an up-down counter. You will need to introduce a signal called- “up_down”. If up_down=1, count up. If up_down=0, count down. Test it with a testbench

Practice - 2

3. Simulate the mux2.v and mux2_tb.v