CSE 390B: Building Academic Success Through Bottom-Up Computing

---

Practice Midterm

Name: _____

UW NetID: _____

Instructions:

- Make sure you have included your name on the exam (first & last) & your student ID #.
- When you are finished with the exam, turn in your exam to the course staff.
- You will have 25 minutes to complete the exam.
- Questions are not necessarily in order of difficulty.
- This exam is closed-note, closed-book (except for the given reference sheet).
- There are 35 points distributed unevenly among 4 questions (most with multiple parts).

Advice:

- Read questions carefully. Understand a question before you start writing.
- Write down thoughts and intermediate steps so you can get partial credit. But clearly indicate what is your final answer.
- The questions are not necessarily in order of difficulty. Skip around. Make sure you get to all the questions.
- If you have questions, please raise your hand and the course staff will get to you shortly.

Question	1	2	3	4	Total
Possible Points	15	6	10	4	35

1. (15 points) Complete the circuit diagram below to implement a reverse two-bit counter which goes through the sequence **11 → 10 → 01 → 00 → 11**. You are provided two DFF gates (A, B) to start, which are directly connected to outputs. For example, if the state is `10`, A=1 and B=0. Assume the DFF gates all start storing 1's, and are implicitly connected to the hardware clock. You may only use And, Or, and Not gates in your implementation.

    a. Here is a template for a truth table that you can fill out to guide your implementation. Fill out the truth table with the correct values based on the transitions described above.

A_t	B_t		A_{t+1}	B_{t+1}
1	1			
1	0			
0	1			
0	0			

    b. We haven't shown you many two output truth tables, but you can essentially think of them as two separate truth tables smushed together (i.e., defining **A**(t+1) based on **A**(t) and **B**(t), and separately defining **B**(t+1) based on **A**(t) and **B**(t)). Using your truth table, define two Boolean expressions, one for **A**(t+1) and one for **B**(t+1), based on **A**(t) and **B**(t).

A(t+1) =

B(t+1) =

c. Implement the circuit below for the logic expressions you wrote above. Your circuit diagram should connect the outputs (**A**(t) and **B**(t)) using logic gates and feed the result back into the corresponding input gates.

2. (6 points) Dana needs 300 pickets for her colorful picket fence. She wants equal amounts of each of her 4 selected colors. She already has 32 red, 26 green, 9 yellow, and no blue. If the pickets cost 25 cents and you get 20% off if you purchase 50 or more of the same color, and 30% off if you purchase 60 or more of one color, how much does Dana need to spend? List your answer to two decimal places. You may use a calculator application on your computer to solve this problem.

3. (10 points) Write a Hack assembly program that stores -1, 0, or 1 in R1 based on the sign of R0. To be more specific, your program should store a -1 in R1 if R0 is negative, a 0 in R1 if R0 is 0, and a 1 in R1 if R0 is positive.

4. (4 points) Name two metacognitive skills that we have covered in CSE 390B so far.