Homework 7
Due: Wednesday, April 4

All homeworks are due at 12:55 PM in the CS22 bin on the CIT second floor, next to the Fishbowl.

Include our cover sheet or equivalent, write your Banner ID (but not your name or your CS login) on each page of your homework, label all work with the problem number, and staple the entire handin before submitting.

Be sure to fully explain your reasoning and show all work for full credit. Consult the style guide for more information.

Problem 1

Cueball has a long backstory with Megan. Because he is reluctant to tell you this story, he chooses five numbers from the set \( \{1, 2, 3, 4, 5, 6, 7\} \) and tells you their product. If you can find out whether the sum of Cueball's numbers is even or odd, he will tell you the story. You realize that the product he gave you is not enough information to tell whether the sum of his numbers is even or odd. What is the product that Cueball told you?

(To discover the backstory, visit comic #<your answer>)

Problem 2

Consider a circuit with \( n \geq 0 \) input wires and one output wire. Let the wires of the circuit be represented by the set \( I_n = \{w_1, w_2, \ldots, w_n\} \), where \( w_i \) represents the \( i \)th wire. Consider some subset \( t \subseteq I_n \). We say that \( t \) satisfies the circuit if the circuit evaluates to true when all input wires in \( t \) are set to true and all input wires not in \( t \) are set to false.

Let \( T_n \) be the set of all \( t \subseteq I_n \) such that \( t \) satisfies the circuit. Prove that for any \( n \geq 2 \), there exists a circuit where:

a. \( |T_n| = 1 \)
b. \( |T_n| = 2^n - 1 \)
c. \( |T_n| = 2^n \)
d. \( |T_n| = 2^{n-1} \)
e. \( |T_n| = n \)
Note: You do not need to use induction or diagram your circuits for this problem. However, you must use every wire in each circuit. Also please do not have more than two inputs into any gates without proving how this will behave.

(For an extra challenge, you may try comic #730)

Problem 3

a. Suppose $p$ and $q$ are propositions such that $p \Rightarrow q$ is False. Determine the truth values of:

(i) $(\neg p) \Rightarrow q$
(ii) $p \lor q$
(iii) $q \Rightarrow p$

b. Design a circuit that takes as input two 1-bit binary numbers $A$ and $B$ and outputs whether or not $A > B$, $A < B$, or $A = B$. Namely, the circuit should have two inputs, the bits $A$ and $B$, and three outputs $G$, $E$, and $L$. For any input, exactly one of $G$ (greater), $E$ (equal), or $L$ (less) should be 1. Note that $G$ corresponds to $A > B$.

(i) Write out a truth table equivalent to this circuit.

(ii) Draw your circuit. Please use only And, Or, and Not gates with at most two inputs per gate. Note that we care only about the correctness of the circuit, not its complexity.

Problem 4

Let $B$ be the set of all possible propositions of any length. Define the relation $R$ over $B$ such that, for $a, b \in B$, $aRb$ if and only if $a$ and $b$ return the same output given the same input.

a. Prove that $R$ is an equivalence relation.

b. Let $x$ be a proposition with the following input/output table:
Write a proposition corresponding to the input/output table given above.

c. The equivalence class of a proposition \( x \) is \([x]_R = \{ a \in B \mid aRx \}\). That is, an equivalence class of proposition is the set of propositions that for every possible input, give the same output.

Prove that \(|[x]_R|\) is infinite for any \( x \in B \).

d. How many equivalence classes of three-input, one-output propositions are there? Explain your answer.

Problem 5

Black Hat has made a nasty habit of stealing self-driving cars. Please help Cueball and Megan come up with an easy way to lock the vehicles! The password will be represented as a three bit integer, using three input wires. When the correct password is entered into the wires, the output wire will activate to unlock the self-driving cars. In order to prevent Black Hat’s from breaking the code, Cueball and Megan need a way to change the password as well. For all parts of the problem, make sure the circuits exhibit the correct functionality in Logisim (See CS0220 Email or Piazza for instructions on testing circuits).

a. Build a circuit in Logisim with three “password entry” input wires for inputting the password, and three more “password definition” wires whose state indicates what the password actually is. When these match pairwise, the password wire should activate.

b. Now copy your circuit and extend it to allow for saving the password persistently. To do this, add a “change password” wire. (This means that there are 7 input wires in total.)

If the change password wire is off, ignore the password definition wires and open the cabinet if and only if the password entry wires match the stored password.
If the change password wire is on, check that the password entry wires match the saved password. If they do, the stored password should become the password currently represented on the password definition wires. If they do not, nothing should happen.

Use D flip-flops (but no clock) and logic gates to model this circuit in Logisim. Note that D flip-flops in Logisim are by default rising-edge. It is not necessary to alter this behavior.

For this problem you may use all logic gates and gates with more than two inputs.