Homework 6
Due: Wednesday, March 21

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

Recall from lecture that \( a \ MOD \ b \) is a function that returns the remainder \( r \) where \( a = qb + r \) where \( q \in \mathbb{Z} \) and \( 0 \leq r < b \).

Danish and Blackhat want to establish a shared secret, while in the presence of an adversary, Beret Guy.

To do this, Danish and Blackhat agree on some large number \( A \), and some prime number \( p \). They then each independently generate a number: Danish generates \( x \) and Blackhat generates \( y \).

After doing this, Danish transmits \( A^x \ MOD \ p \) to Blackhat over a public channel, and Blackhat transmits \( A^y \ MOD \ p \) to Danish.

Blackhat generates the shared secret by computing \( (A^x \ MOD \ p)^y \ MOD \ p \), and Danish generates the shared secret by computing \( (A^y \ MOD \ p)^x \ MOD \ p \). Doing this gives both of them the shared secret, \( A^{xy} \mod p \).

Beret Guy, who has been listening to their communications over the public channel, knows \( A \), \( p \), \( (A^x \ MOD \ p) \) and \( (A^y \ MOD \ p) \). However, since he doesn’t know \( x \) or \( y \), it is very hard for him to calculate the shared secret.

Show that \( (A^x \ MOD \ p)^y \ MOD \ p = (A^y \ MOD \ p)^x \ MOD \ p \), thereby allowing Blackhat and Danish to calculate the shared secret.

Note: it may be helpful when starting this problem to try this out with actual numbers, in order to better understand what you need to show.
Problem 2

For each of the following, answer the questions and provide your reasoning.

a. Can AND (\(\land\)) be expressed using only OR (\(\lor\)) and NOT (\(\neg\))?

b. A set of Boolean operations is xkcd approved if all possible truth tables can be expressed using only the operators in that set. Is the set \{OR, AND\} xkcd approved?

c. Is the set \{OR, AND, NOT\} xkcd approved?

**Hint:** Consider the truth table representing an arbitrary boolean function. Is there a way that you could express this table in terms of the above operators?

d. Consider the \(\star\) operator, defined by \(p \star q = \neg(p \land q)\). Is the set \{\(\star\)\} xkcd approved?

Problem 3

Cueball came up with a diabolical brain teaser in one of his dreams, but he forgot the answer when he woke up! Help him remember so that he can boggle his friends’ minds.

Which of the following is correct? There is exactly one answer.

a) none of the below
b) none of the below
c) one of the below
d) all of the below
e) none of the above
f) all of the above

(For another diabolical but funnier brain teaser visit comic #1134) **Hint:** Use a truth table!

Problem 4

Hairy and Ponytail have been imprisoned in separate comics for unknown crimes. As a cruel and unusual punishment, they must each flip a coin every morning and guess the result of the other prisoner’s coin flip. They can’t communicate with each
other at all, and only know the result of their own coin flip. If at least one of the
two guesses is correct, they will live another day; if not, they will be erased.

Before being locked away, they have time to decide on a plan to spare themselves
indefinitely. How should they guess on each day? **Hint**: Use a truth table.

**Problem 5**

a. In Logisim, you can create **And** and **Or** gates that take in more than two inputs.
Under the hood, these can just be implemented with 2-input **And** and **Or** gates.
In Logisim, build a 3-input **And** circuit using only 2-input **And** gates, and build
a 3-input **Or** circuit using only 2-input **Or** gates.

b. Cueball has spent many election cycles closely following poll data and the words
of Nate Silver. He decides that his time digesting information is over: he wants
to run a poll himself! However, to do so, he needs a machine that can handle the
poll data.

To help him out, construct a circuit in Logism using only **And**, **Or**, and **Not**
gates. You may use gates of any number of inputs for this problem.

The circuit will take in three people's answers (1 for yes and 0 for no), and return
the answer of the majority.

(To find out about Cueball’s obsession, visit comic #1130)

c. Cueball is currently working on the Collatz conjecture, and is having a lot of
trouble determining if the numbers 0, 1, 2, and 3 are odd or even. The solution?
Build a circuit!

Model this problem as a circuit in Logisim using only **And**, **Or**, and **Not** gates.
You may use gates with more than one input. Create three input wires and one
output wire. If an even number of inputs are on, the output should be off. If an
odd number of input wires are on, the output should be on.

Note: The output depends only on how many of the input wires are on. For
example, if exactly one input is on, then the output should be on. It does not
matter which input is on.

(Let’s hope he solves it! Visit comic #710)