Computer Science Foundations
Puzzle-Solving Workshop and Seminar

Episode 3—October 14
Fall 2013

Paul Pham and Neal Nelson

October 11, 2013

Welcome to the Puzzle-Solving Workshop and Seminar for Computer Science Foundations. You will notice there are two parts to this thread: *workshop* and *seminar*. Workshop is meant to strengthen your problem-solving skills, to help you in the Discrete Math thread as well as future math and computer science courses. Seminar is meant to work on your discussion, writing, and creativity skills. We will alternate between the two different modes.

Today we will be in workshop mode only. We will go over the solutions to last week's problems (Gauss's problems and Ada's problems). We will teach other in a special group exercise that doubles in size each time and finally includes the entire class.

The following workshop problems are taken from Sherri Shulman's Discrete Math Workshop on October 8, 2012.

1 Problem 1: Logic Symbols

Ada and Carl Friedrich (Gauss) decide to play a game by only speaking in propositional logic variables. They define the following variables to stand for English sentences about the weather, and then they will use them to reason about facts about the weather.

c	=	"It is cloudy."
r	=	"It is raining."
s	=	"It is sunny."
w	=	"It is night."

Ada comes up with three rules, or things that she believes to be true about the weather.

- If it is cloudy, it cannot be sunny.
- If it is not sunny, then it is either cloudy or it is nighttime.
- If it is rainy, the ground is wet and it is cloudy.

(a) Translate each of Ada's rules above into propositional logic, using the variables c, s, n, and r combined with the logical operators $\wedge$, $\vee$, and $\neg$.

In response, Carl also comes up with three rules, or things that she believes to be true about the weather.

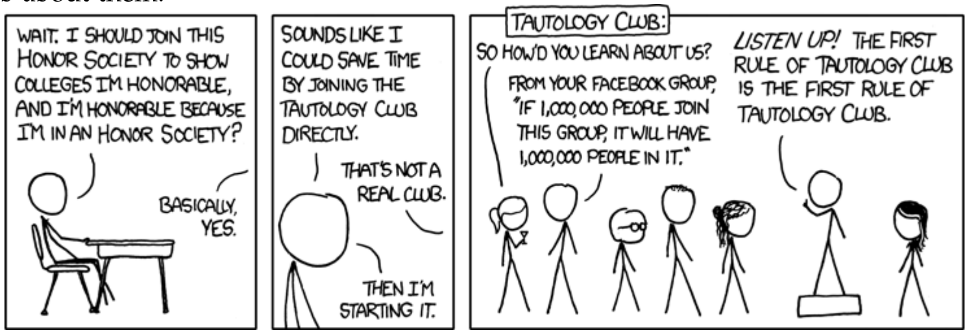
- It is raining when the ground is wet.
- If it's sunny, then its not raining.
- It only rains at night.

(b) Likewise, translate all of Carl's rules into propositional logic.

2 Problem 2: Tautologies, Contradictions, and Everything in Between

Ada and Carl notice a curious thing about logic. Some propositional logic expressions are true no matter what. If you were to draw out their truth table, they would always evaluate to True. These are called *tautologies*. It doesn't matter what the variables stand for. They could be sentences about weather like in the previous example, or anything else, or completely meaningless symbols.

Ada and Carl are so excited about tautologies that they decide to form a club about them.



Moreover, there are logical expressions that are always false no matter what. These are called *contradictions*. Then there are expressions which are true or false depending on their input variables. Most logical expressions fall in this category.

(a) Create a truth table for each of the following expressions. Determine whether they are always true (a tautology), always false (a contradiction), or neither.

- $(p \vee q) \wedge (\neg p \vee q)$
- $p \rightarrow (\neg p \rightarrow q)$

- $(\neg(p \wedge q)) \leftrightarrow (\neg p \vee \not q)$

In their tautology club, Ada and Carl spend a lot of time separating all the logical expressions in the world into three categories: tautologies, contradictions, and everything else. To do this, they must decide whether two expressions are logically equivalent, and they need your help.

(b) Ada and Carl need to know whether $p \rightarrow q$ and $\neg q \rightarrow \neg p$ are logically equivalent. Ada likes to use truth tables, because they seem sure and foolproof, while Carl likes to take shortcut and use logical symbols and operators.

Are these two expressions equivalent?

- First use Ada's method and write out the truth tables for both expressions.
- Then use Carl's method and try to transform one expression into the other using logical equivalence rules that you learned in class.

(b) Carl tries to convince Ada that truth tables take much too long for even small numbers of boolean variables. Without using truth tables, show that $\neg p \rightarrow (q \rightarrow r)$ and $q \rightarrow (p \vee r)$ are logically equivalent. How many rows would the corresponding truth table have for each expression?

(b) Ada believes that truth tables are the best way to represent logical expressions because they are complete and visual. Everything is laid out in neat rows for you to see, and they are always the same size. Logical expressions are sometimes more compact, but some expressions are just as long as their truth tables! Furthermore, it's hard to know whether you've found the shortest or the *best* logical expression, since it can take on multiple equivalent forms. She challenges Carl to find a logical expression corresponding to the following truth table, and he needs your help.

p	q	r	??
T	T	T	T
T	T	F	F
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	T
F	F	T	F
F	F	F	F

(b) **Optional** Help convince Ada that logical expressions can often be useful and more compact than their corresponding truth tables. Generalize your method in the previous problem to apply to any number of variables, for any truth table. This is what we mean when we say that $\neg$, $\wedge$, and $\vee$ are a complete (universal) set of logical connectives.

3 Reading for Discussion Next Time

Read the following chapter from Lauren Ipsum for next Monday's workshop/seminar: "A Tinker's Trade" at `http://www.laurenipsum.org/tinker`. Come prepared to discuss algorithms, loops, notions of "sameness" and "differentness" between two pieces of code, and how to combine two different pieces of code into the same one.