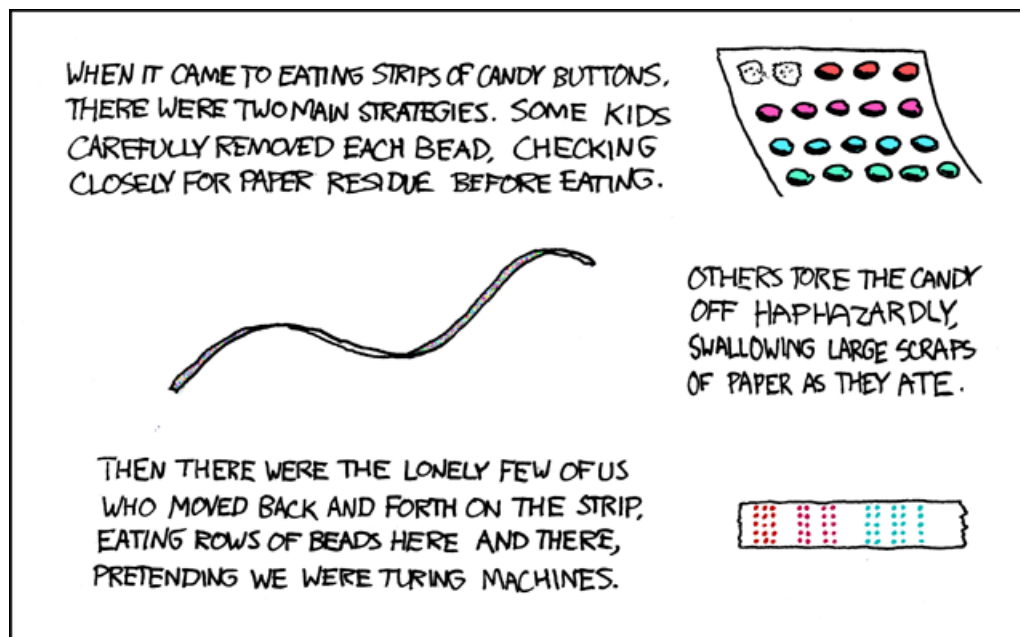# CS30 - Assignment 11

Due Sunday, May 3 at 11:59pm

https://xkcd.com/205/

For this assignment you will be playing with the JFLAP automata tool to design DFAs and turing machines.

Obtaining JFLAP

We will use a Java-based program, called JFLAP, that we've been playing with in class. It was written by Professor Susan Rodger of Duke University and you can obtain it at:

http://www.jflap.org/

Click on the tab to the left "Get JFLAP", fill out the form and then download version 7.0 (NOT the beta version, 8.0). The "Thin" version should work, but if it doesn't, try downloading the full version. Once you download the .jar file you *should* only need to double-click on it for it to run.

Even if you don't plan on doing the assignment early, make sure to download JFLAP ASAP and make sure that it works on your computer.

If you need help using JFLAP, select the "Help" menu from the program or you can view a tutorial online at `http://www.jflap.org/`.

DFAs

For each problem below, save your JFLAP machine in a file named

`asgt11-⟨one-digit-problem-number⟩.jff`

1. [**4 points**] Use JFLAP to construct a DFA that accepts all strings that have exactly one `a` and two or more `b`'s.

2. [**4 points**] Use JFLAP to construct a six-state DFA which accepts a string if it has a number of `a`'s that is a multiple of 2 or a multiple of 3 (or both). You can assume that the language is only over a's (i.e. no b's)

3. [**5 points**] Consider strings over the alphabet with two letters, 0 and 1. Use JFLAP to construct a DFA that accepts all strings which are the binary representations of multiples of 3.

Turning machines

4. [**5 points**] Construct a turing machine over the input alphabet of parentheses, i.e.) and (, that recognizes the language of all strings of properly balanced parentheses (equal number of open and close parentheses *and* the open parentheses must come before the corresponding close parentheses). For example

()
(())
(()((()())))

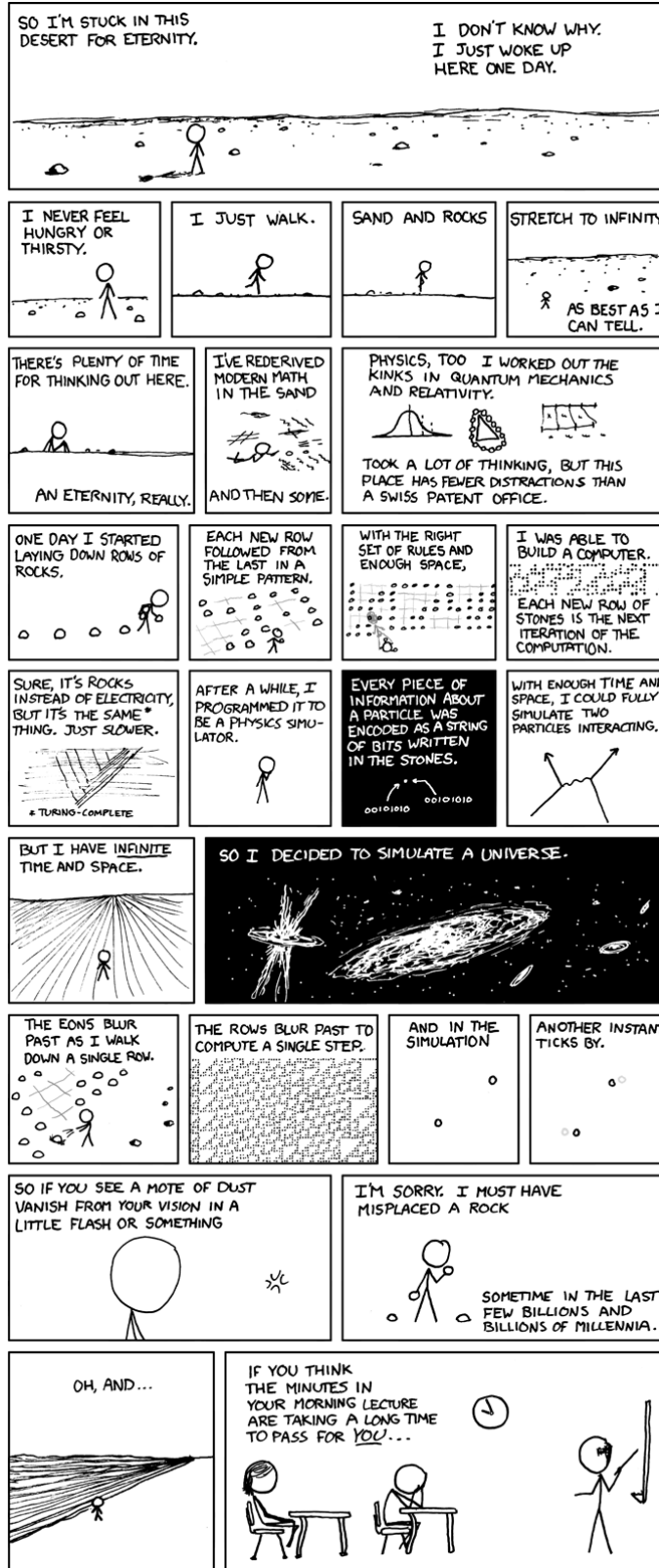would all be members of the language, while

)(
(()
())(

would NOT be members of the language.

Big Hint: Recall the Turing Machine that we built in class which accepts all strings with an equal number of 0's and 1's. The only major difference between these two problems is that in the 0's and 1's problem, it didn't matter which occurred first, whereas with this problem, the open paren needs to occur first in the input string.

When you're done

You should have a four .jff files. Put all of these files in a folder named `assign11-` followed by your first and last name. `zip` up this folder and submit it via the online course submission mechanism.

Make sure that you follow these instructions exactly.