CS 7A - Spring 2016 - Elementary Cellular Automata II. Due 5/5/16

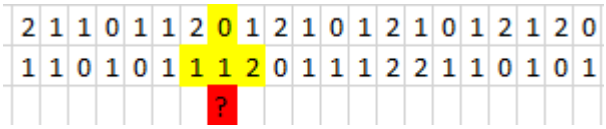
Modify the one-dimensional cellular automata program in the following ways:

1. Use a `CA` class that includes data member `int gen` to keep track of how many generations have been produced, `vector<vector<int> > gens` to hold each of the generations and a member function `vector<int> generate(vector<int> vi)` to produce the next generation from the previous generations.

2. Increase the number of states a cell can be in from 2 to 3.

3. Increase the neighborhood of a cell from 3 to 4 (left, middle, right and previous) so there are $3^4 = 81$ rules to follow. Obviously you don't want to right a separate line of code for each rule, so you could could write a loop counting from rule $0 = 0, 0, \ldots, 0$ to rule $81 = 2, 2, \ldots, 2$ or you could set rules randomly as you need them.

   What you want is to set things up so that you can look up the rule for deciding the state of a cell based on the 4 cells in its neighborhood, say a situation like this:

   where the left, middle, right and previous cells to the red question mark square to be generated are 1,1,2,0. So you need a function to look up that rule and see what result (0,1 or 2) has been assigned to it. There are 81 rules which you can set up any way you like, even randomly, as we described in class.

4. Write a few paragraphs about the dynamics you observe for various initial states and rules.