CS 383
Exam 1

1. Which languages are regular? You don't need to prove your answer. Write an "R" in the blank next to the description of each language you think is regular. Write "N" for any language you think is not regular. In each case the alphabet is Σ={0,1}

    a. _R____Strings where the number of 0's and the number of 1's are either both even or both odd.

    b. __N___Strings of odd length with 1 in the center, such as 0001000 or 1101100

    c. __R__Strings of the form $\alpha\beta$ where $|\alpha| = |\beta|$ such as 000111

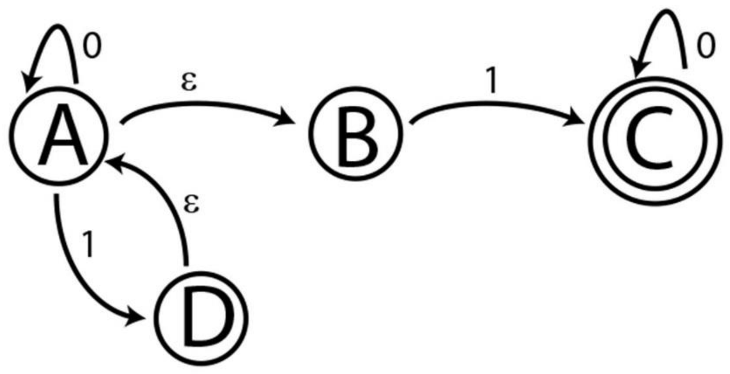
    d. __R___Strings that start and end on the same digit, such as 1010101 and 0001110

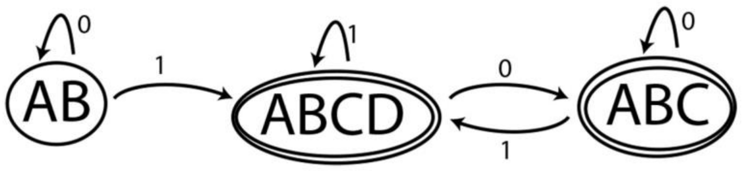
    e. __N__Strings that have the same numbers of 0's and 1's

    f. __R___Strings whose digits sum to no more than 1000.

    Note that languages (a) and (c) are the same; these are both fancy descriptions of strings of even length.

2. Here is an ε-NFA, with start state A. Convert this NFA to a DFA and describe in English the strings it accepts.





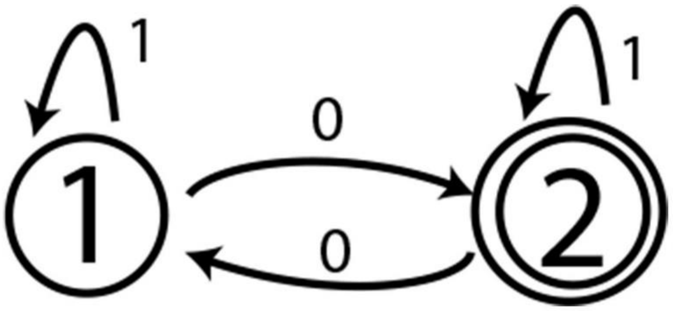
Note that an string that ever gets to the state ABCD is accepted. This automaton accepts all strings that contain 1.

3. Find an ε-NFA that accepts the language described by the regular expression

$$1(00+11)^{*}01$$

This would take me a long time to draw, so ask me if you want to see it. The only tricky part is making sure that you can do (00+11) 0 times.

4. Consider the following DFA. We had an algorithm for converting a DFA to a regular expression. This involved making a table of regular expressions r_{ij}^{k} .



Here is the first column of a table of the r_{ij}^{k} expressions; find the 4 entries of the second column.

	k=0	k=1
r_{11}^{k}	ε+1	1*
r_{12}^{k}	0	1*0
r_{21}^{k}	0	01*
r_{22}^{k}	ε+1	ε++1+01*0

5. Here’s a definition of a regular expression over the alphabet Σ:

   a. ε and Ø are regular expressions
   b. Any string in Σ^* is a regular expression
   c. If E and F are regular expressions then so are E+F, EF, and E*.

   Use Structural Induction to show that any regular expression that does not use the * operator describes a finite language.

   Base cases: e and Ø represent languages{e} and Ø, which are finite.
   If w is a string in Σ^* w represents {w}, which is finite.

   Inductive cases: suppose E and E are regular expressions to which the result applies. If E+F doesn’t use the *operator then neither do E or F so they represent finite languages, say with n and m strings. E+F represents the union of those languages, with no more than n+m strings, so it is finite. Similarly EF represents the concatenation of the languages represented by E and F, with no more than n*m strings, so it also is finite. So any expressions you can build out of E and F without using the * operator represent finite languages.

6. Give a careful proof of the fact that the language $\{0^n1^m \mid n > m\}$ is not regular.

   This is a proof by contradiction. Suppose the language is regular and let n be its pumping constant. Consider the string $w = 0^n1^{n-1}$. The Pumping Lemma says there is a decomposition of w, w=xyz, where $|xy| < n$ and y is not the empty string such that xy^kz is in the language for all k. However, since $|xy| < n$ and the first n characters of w are all 0, y consists of a positive number of 0s. xy^0z is not in the language since it does not have more 0s than 1s. This violates the Pumping Lemma, so the language is not regular..