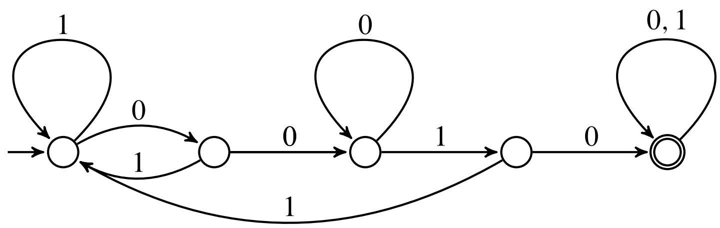
CS 181A EXAM #1 **NAME** ______________________________

SPRING 2014 **UCLA ID** ______________________________

You have 90 minutes to complete this exam. You may assume without proof any statement proved in class.

(3 pts) **1** Give a simple verbal description of the language recognized by the following DFA.

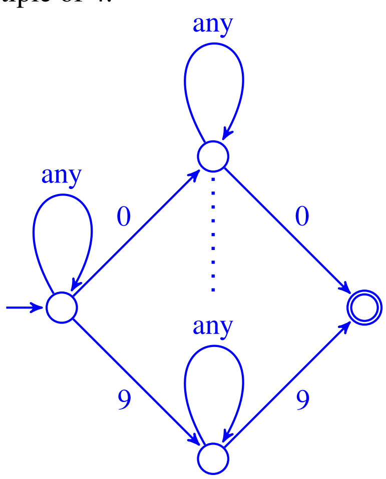


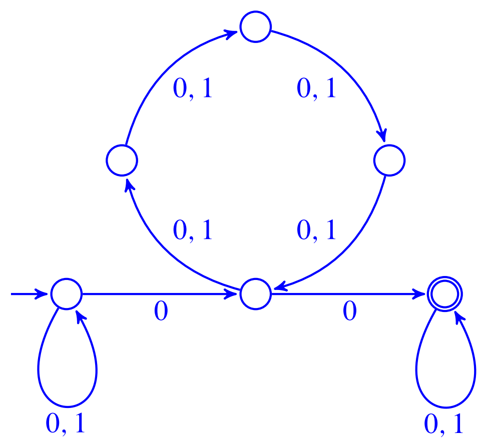
Solution. All binary strings that contain 0010.

2 Draw NFAs for the following languages, taking full advantage of nondeterminism:

(2 pts) **a.** strings over alphabet $\{0, 1, \ldots, 9\}$ where the final digit has appeared before;

(2 pts) **b.** binary strings in which there is a pair of 0s separated by a number of positions that is a multiple of 4.





3 Prove that the following languages over the binary alphabet are regular:

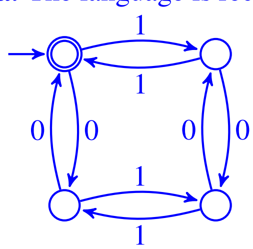
(2 pts) **a.** strings in which the number of 0s and the number of 1s are both even;

(2 pts) **b.** strings with at most one occurrence of the substring 00 (the string 000 has two);

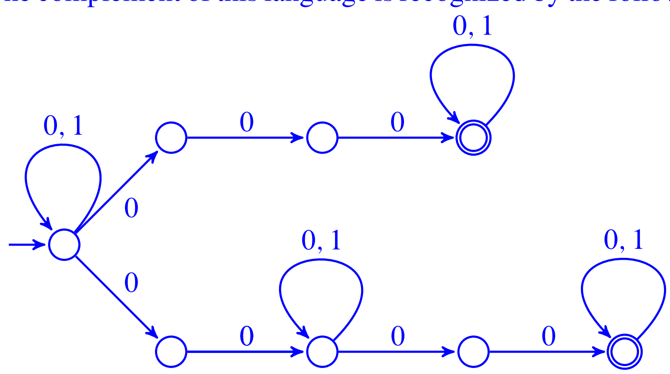
(2 pts) **c.** strings in which the 1000^{th} symbol from the end is a 1.

Solution.

a. The language is recognized by the following DFA:



b. The complement of this language is recognized by the following NFA and is therefore regular:



Since regular languages are closed under complement, the original language is regular as well.

c. To recognize this language, a DFA simply needs to keep track of the last 1000 symbols seen, and accept if and only if there is a 1 in position 1000. Formally, the language is recognized by the DFA

$$(\{0,1\}^{1000}, \{0,1\}, \delta, 0^{1000}, 1\{0,1\}^{999}),$$

where $\delta(w_1 w_2 w_3 \ldots w_{1000}, \sigma) = w_2 w_3 \ldots w_{1000}\sigma$.

(3 pts) **4** Let L be a regular language over the binary alphabet. Consider the following language over the same alphabet: $L' = \{w : |w| = |u| \text{ for some } u \in L\}$. Prove that L' is regular.

Solution. To obtain an NFA for L', start with a DFA for L and change all edge labels to "0, 1".

(3 pts) **5** Prove that at most $k^{2k+1}2^k$ languages over the binary alphabet can be recognized by a DFA with k states.

Solution. Simply count the number of distinct DFAs with k states. Name the states $1, 2, 3, \ldots, k$. Then a DFA is a tuple

$$(\{1, 2, 3, \ldots, k\}, \{0, 1\}, \delta, q_0, F),$$

where

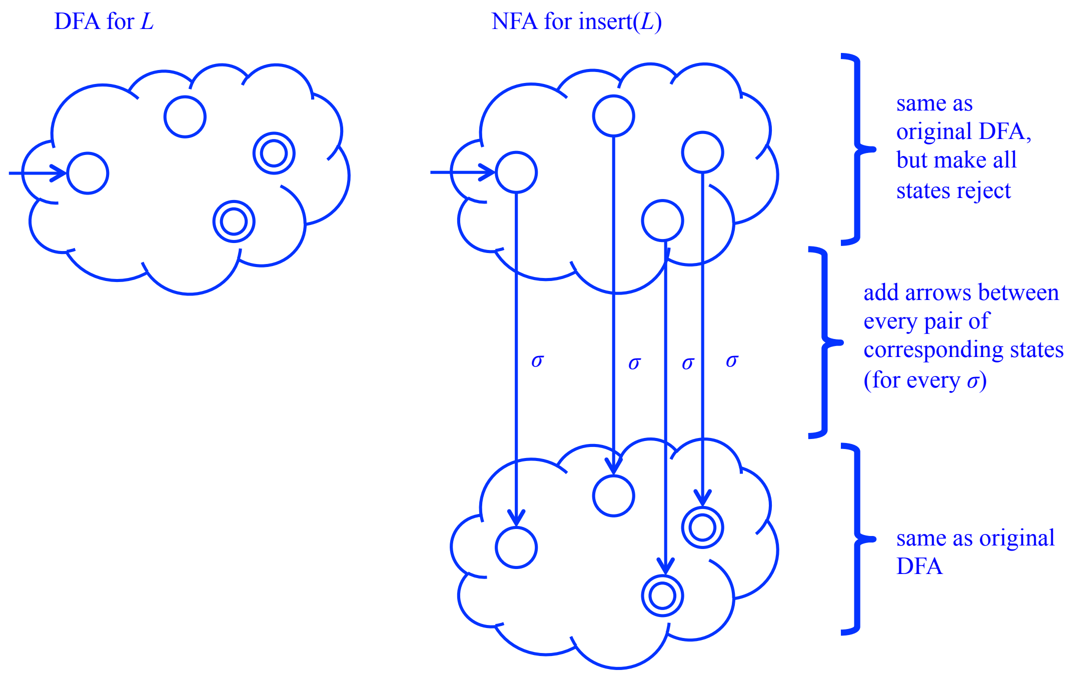
$$q_0 \in \{1, 2, 3, \ldots, k\}$$
$$F \subseteq \{1, 2, 3, \ldots, k\},$$
$$\delta : \{1, 2, 3, \ldots, k\} \times \{0, 1\} \to \{1, 2, 3, \ldots, k\}.$$

Thus, the number of distinct ways to choose (q_0, F, δ) is

$$k \times 2^k \times k^{2k}.$$

(3 pts) **6** For a language $L \subseteq \Sigma^*$, define insert$(L) = \{u\sigma v : uv \in L, \sigma \in \Sigma\}$. Thus, insert$(L)$ is the set of all strings obtained by taking a string in L and inserting a new character at some position. Prove that if L is regular, so is insert(L).

Solution:



(3 pts) **7** For a language L, define suffix$(L) = \{v : uv \in L$ for some $u\}$. Thus, suffix(L) is the set of all suffixes of strings in L. Use the closure of regular languages under the reverse and prefix operations to prove that suffix(L) is regular whenever L is regular.

Solution. To generate the suffixes of all strings in L, one can reverse the strings in L, generate all prefixes in the resulting language, and finally reverse the resulting strings. Thus,

$$\text{suffix}(L) = \text{reverse}(\text{prefix}(\text{reverse}(L))).$$

Since L is regular and regular languages are closed under the prefix and reverse operations, it follows that suffix(L) is regular.