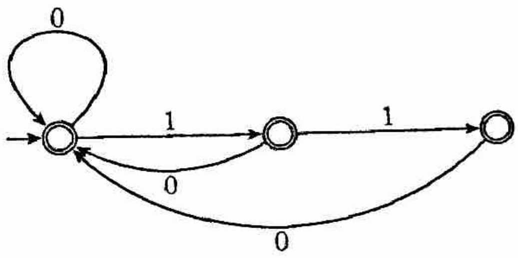
CS 181 EXAM #1 NAME NIKHIL KANSAL
FALL 2018 UCLA ID —

20

You have 90 minutes to complete this exam. You may state without proof any fact taught in class or assigned as homework.

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



All binary strings where $\Sigma = \{0,1\}$ that contain at most two 1's in a row.

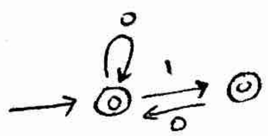
+3

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

(2 pts) **a.** binary strings in which every pair of 1s are separated by at least one 0;

(2 pts) **b.** binary strings that have odd length or do not contain a 0.

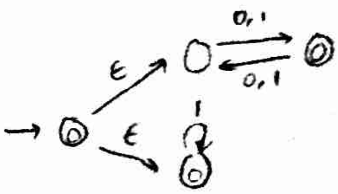
a.



+2

(= does not contain "11")

b.



+2

3 Prove that the following languages are regular:

(2 pts) a. binary strings whose length is a prime number less than 100;

(2 pts) b. binary strings with precisely one occurrence of 01 as a substring;

(2 pts) c. binary strings of odd length in which the first and last symbols are not the same.

a. Construct an NFA $N = (Q, \Sigma, \delta, q_0, F)$

$$Q = \{ q_i \mid 0 \le i < 100 \}$$

$$\Sigma = \{0, 1\}$$

(N accepts the language, so the language is regular)

+2

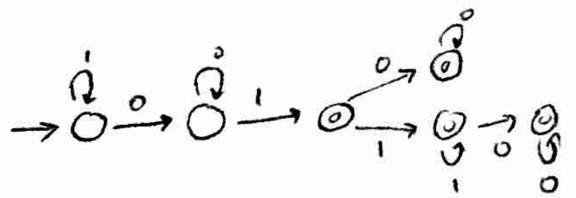
$$\delta(q_i, w) = \begin{cases} \{q_{i+1}\} & \text{if } i < 99 \text{ and } w \in \Sigma \\ \emptyset & \text{otherwise} \end{cases}$$

$$q_0 = q_0$$

$$F = \{ q_i \mid i \text{ is a prime number less than } 100 \}$$

$(2, 3, 5, 7, 11, 13, 17, 19, \ldots, 97)$

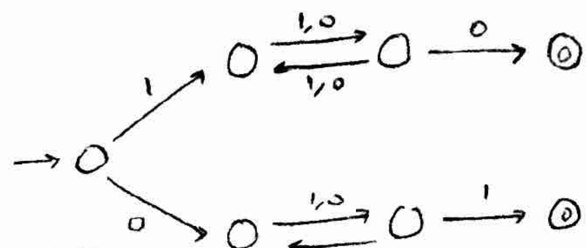
b.



(NFA exists, so language is regular)

+2

c.



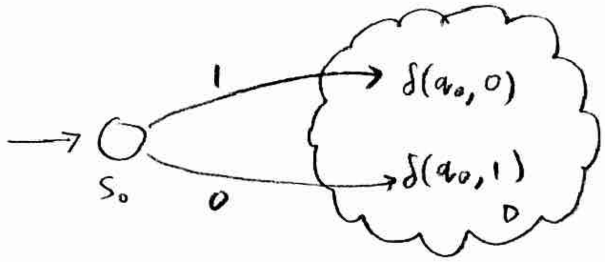
(NFA exists, so language is regular)

+2

(3 pts) **4** Let L be a regular language over the binary alphabet. Let L' be the set of strings obtained by taking a nonempty string in L and flipping its first bit. Prove that L' is regular.

3

Since L is a regular language, it has a DFA $D = (Q, \Sigma, \delta, q_0, F)$ that accepts it. We can create a new NFA from D that accepts L':



- create a new start state S_0
  - On a 1 transition, transition into the DFA D as if it had received a 0 from its original start state
  - on a 0 transition, transition into the DFA D as if it had received a 1 from its original start state.

(3 pts) **5** Your friend draws a DFA for you and colors a subset of its states purple. Let L be the set of strings v such that the DFA never reaches a purple state while processing v. (For example, coloring the start state purple forces $L = \varnothing$.) Prove that L is regular.

Suppose the DFA we are given is D. We can modify D to create an NFA N that accepts L.

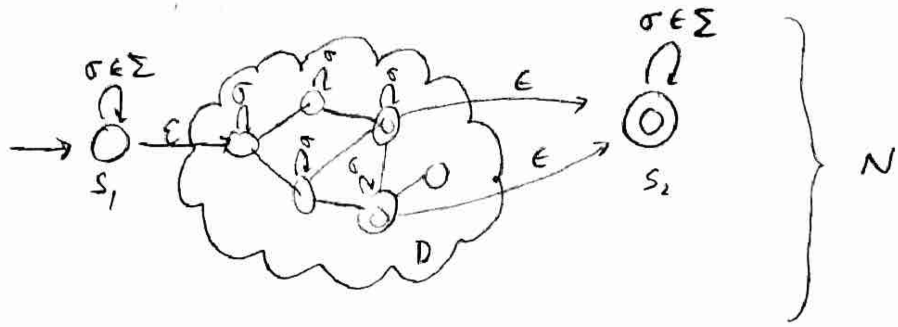
For each purple state in D, remove all inbound transitions to that state. This means a purple state can never be reached while processing a string in N. This new NFA now recognizes L.

- start state
- non purple → accept

3

(3 pts) **6** For a string v, a *supersequence* of v is either v itself or any string obtained by inserting one or more characters into v. For example, `transit`, `site`, `snippet` are supersequences of `sit`, but `it` is not. Prove that for any regular language L, the set of all supersequences of strings in L is a regular language.

Since L is a regular language, there exists a DFA D that accepts L. We can construct an NFA N from D that accepts supersequence(v):



(where Σ is the alphabet)

1.) create new start state s_1 that transitions to itself on all $\sigma \in \Sigma$ and epsilon transitions to the start state of D. (inserts an arbitrary prefix)

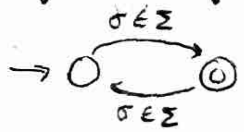
2.) Add loop transitions for each $\sigma \in \Sigma$ for each state in D that is on a path to an accept state. (inserts chars into the string)

3.) Create a new accept state s_2 outside D. Create epsilon transitions from each accept state in D to s_2. Create a loop transition on s_2 for every $\sigma \in \Sigma$. (inserts an arbitrary suffix)

3

(3 pts) **7** Fix a regular language L. Consider the language L' of strings of the form $w_1 w_2 \ldots w_{2018}$, where $w_1, w_2, \ldots, w_{2018}$ are odd-length strings in L. Prove that L' is regular.

Let A be the set of all odd-length strings. A is regular because it has an NFA:



(where Σ is the alphabet)

Let B be the set of all odd-length strings in L, i.e. $B = L \cap A$. By closure property proved in class for set intersection, B is regular.

Strings in L' are the concatenation of strings in B:

$$L' = \{w_1 \circ w_2 \circ \ldots \circ w_{2018} \mid w \in B\}.$$

Since each w_i and $w_j \in B$ are strings in a regular language, $\{w_i w_j \mid w_i, w_j \in B\}$ is regular. We can iteratively apply the closure property for string concat to see that $L' = \{w_1 \circ w_2 \circ \ldots \circ w_{2018} \mid w \in B\}$ is thus also regular. Thus, L' is regular.