CMSC 303 Introduction to Theory of Computing, VCU
Assignment 3

Turned in electronically in PDF, PNG or Word format before the start of class

Key

Total marks: 62 marks + 3 bonus marks for all the answers typed out.

Unless otherwise noted, the alphabet for all questions below is assumed to be $\Sigma = \{0, 1\}$.

1. [12 marks] This question develops your ability to devise regular expressions, given an explicit definition of a language. For each of the following languages, prove they are regular by giving a regular expression which describes them. Justify your answers.

   (a) $L = \{x \mid x$ begins with one 1 and ends with two 0's$\}$.
   Solution: $R = 1\Sigma^*00$. This regular expression describes L since any string described by R must start with an 1 and end with two 0's, and can have any other characters in between.

   (b) $L = \{x \mid x$ contains at least three 0's$\}$.
   Solution: $R = \Sigma^*0\Sigma^*0\Sigma^*0\Sigma^*$. This regular expression describes L since any string described by R must contain at least three 0's. These 0's can be surrounded by any strings desired.

   (c) $L = \{1, 111, \epsilon\}$.
   Solution: $R = 1 \cup 111 \cup \epsilon$. Since L has a finite number of elements, to obtain a regular expression for it, we can simply take the union of a finite number of strings, each representing a distinct element of L.

   (d) $L = \{x \mid$ the length of x is at most $5\}$.
   Solution: $R = (\Sigma \cup \epsilon)(\Sigma \cup \epsilon)(\Sigma \cup \epsilon)(\Sigma \cup \epsilon)(\Sigma \cup \epsilon)$. This regular expression allows you to pick either 0, 1, 2, 3, 4 or 5 characters from Σ in any order, as required for L.

   (e) $L = \{x \mid x$ doesn't contain the substring $bba010\}$.
   Solution: $R = (0^+11 \cup 1)^*(0^+1 \cup 0^+ \cup \epsilon)$. Our justification is created by first creating a DFA that recognizes 010, then converting that to a DFA that does not recognize 010. Then converting that to a generalized NFA, and converting that to a Regular expression following Lemma 1.60.

   (f) $L = \{x \mid |x| > 0$, i.e. x is non-empty$\}$.
   Solution: $R = \Sigma^+$. This regular expression ensures each string has at least one character from Σ, as required by L.

2. This question tests your understanding of how to translate a regular expression into a finite automaton. Using the construction of Lemma 1.55, construct NFAs recognizing the languages described by the following regular expressions.

(a) [5 marks] $R = \emptyset^*$.
Solution:

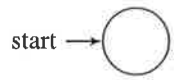


Figure 1: $\emptyset$

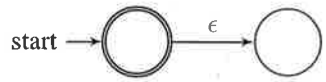
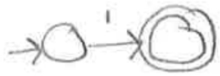


Figure 2: $\emptyset^*$

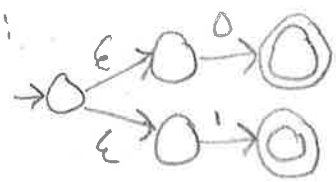
(b) [10 marks] $R = (0 \cup 1)^*010(0 \cup 1)^*$.
Solution:

0:

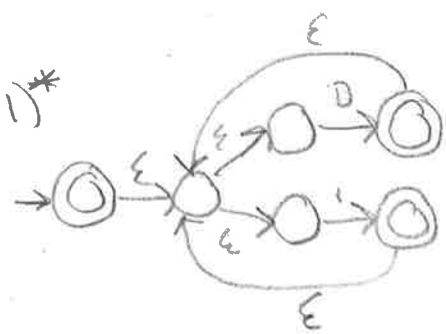
1:



$0 \cup 1$:

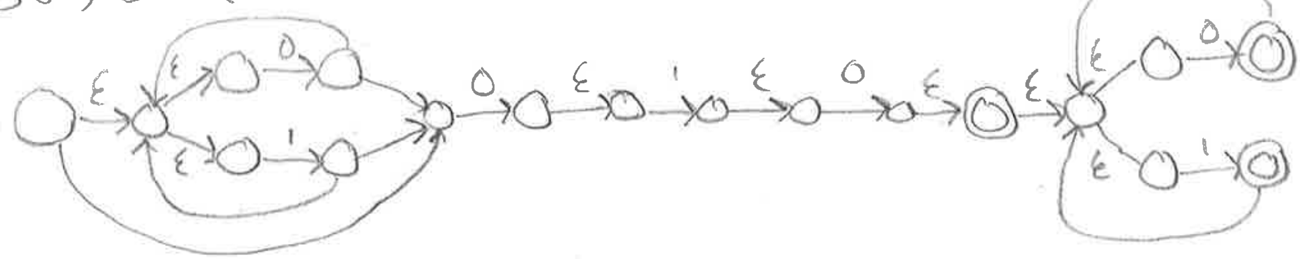


$(0 \cup 1)^*$



010:

$(0 \cup 1)^* 010 (0 \cup 1)^*$



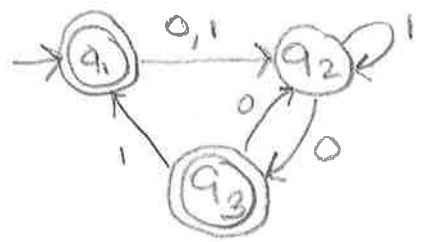
3. [15 marks] This question tests your understanding of how to translate a finite automaton into a regular expression. Consider DFA $M = (Q, \Sigma, \delta, q, F)$ such that $Q = \{q_1, q_2, q_3\}$, $q = q_1$, $F = \{q_1, q_3\}$, and δ is given by:

δ	0	1
q_1	q_2	q_2
q_2	q_3	q_2
q_3	q_2	q_1

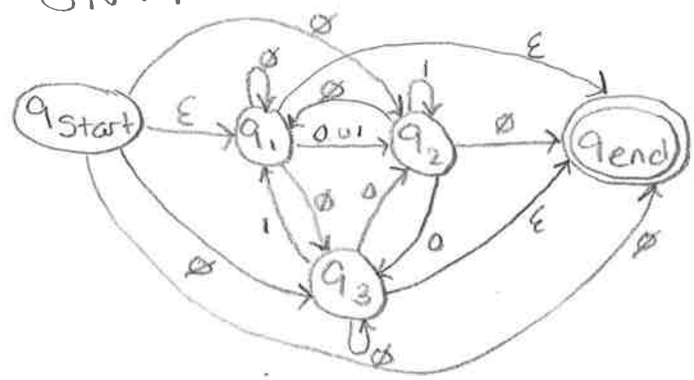
Draw the state diagram for M, and then apply the construction of Lemma 1.60 to obtain a regular expression describing $L(M)$.

Solution:

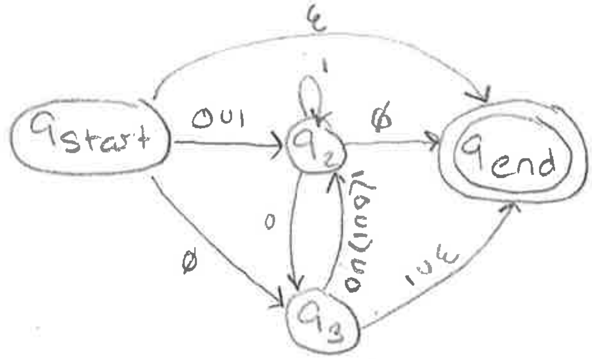
DFA



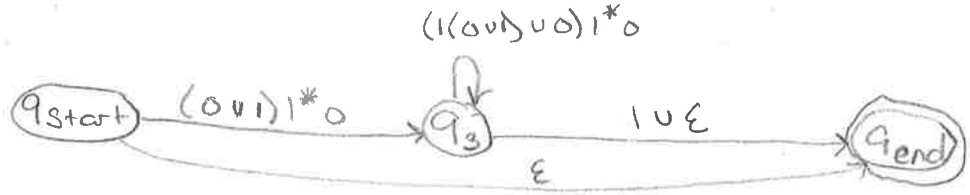
GNFA



$q_{rip} = q_1$



$q_{rip} = q_2$



$q_{rip} = q_3$

$$q_{start} \xrightarrow{((0 \cup 1)1^*0)((1(0 \cup 1) \cup 0)1^*0)^*(1 \cup \varepsilon) \cup \varepsilon} q_{end}$$

4. [10 marks] This question allows you to practice proving a language is non-regular via the Pumping Lemma. Using the Pumping Lemma (Theorem 1.70), give formal proofs that the following languages are *not* regular:

   (a) $L = \{www \mid w \in \{0,1\}^*\}$.
   Solution: We proceed by contradiction. Assume L is regular with pumping length p. Consider string $s = 0^p10^p10^p1 \in L$. The Pumping Lemma now states that there exists a decomposition $s = xyz$ such that $|xy| \leq p$, $|y| > 0$, and y can be pumped. By the first two of these conditions, we know that y must comprise some non-empty substring of the first p zeroes in s. Hence, by choosing arbitrarily large $i > 1$, the string $s' = xy^iz$ can be made to begin with an arbitrarily large substring of zeroes, which implies $s' \notin L$, since the number of zeroes between 1's in s' remains fixed at p no matter how large i gets. But the Pumping Lemma claims that $s' \in L$. Thus, we have a contradiction.

   (b) $L = \{1^n0^m1^n \mid m, n \geq 0\}$.
   Solution: We proceed by contradiction. Assume L is regular with pumping length p. Consider string $s = 1^p01^p \in L$. The Pumping Lemma now states that there exists a decomposition $s = xyz$ such that $|xy| \leq p$, $|y| > 0$, and y can be pumped. By the first of these two of these conditions, we know that y must comprise some non-empty substring of the first p ones in s. Hence, the string $s' = xy^2z = 1^{p'}01^p$ for $p' > p$, implying $s' \notin L$. But the Pumping Lemma claims $s' \in L$. Thus, we have a contradiction.

5. This question further tests your understanding of the subtleties of the Pumping Lemma:

   (a) [5 marks] Let $B_1 = \{1^ky \mid y \in \{0,1\}^*$ and y contains at least k 1s, for $k \geq 1\}$. Show that B_1 is a regular language.
   Solution: We can let $B_1 = 1\Sigma^*1\Sigma^*$, which is regular.

   (b) [5 marks] Let $B_2 = \{1^ky \mid y \in \{0,1\}^*$ and y contains at most k 1s, for $k \geq 1\}$. Show that B_2 is not a regular language.
   Solution: We can show that B_2 is non-regular using the pumping lemma. Assume B_2 is regular and let $p = xyz$ satisfying the three conditions. Condition three says that y appears among the left-hand 1s. We pump down to obtain the string xz which is not a member of B_2. Therefore B_2 does not satisfy the pumping lemma and hence is not regular.