HW1

Due: September 21, 2021

Reminder: Your name should not appear anywhere on your handin; each individual page of the homework should include your Banner ID only. For your digital submission, each page should include work for only one problem (i.e., make a new page/new pages for each problem).

While collaboration is encouraged in this class, please remember not to take away notes from any labs or collaboration sessions, and to list your collaborators at the beginning of your homework submission.

Please monitor Piazza, as we will post clarifications of questions there. Hand in your solutions by 10:20am to Gradescope. See the course syllabus for the late policy.

Problem 1

Describe the language recognized by the following DFA:

Solution:

\{ w | \text{every odd position of } w \text{ is a 1} \}

Problem 2

Design a DFA for each of the following languages under the alphabet \{0, 1\}, and provide a simple proof of the correctness of each DFA:
(a) \( \{ w_A \mid w_A \text{ begins with 0 and ends with 1} \} \)

(b) \( \{ w_B \mid \text{the length of } w_B \text{ is at most 2} \} \)

(c) \( w_A \cup w_B \). Also describe \( w_A \cup w_B \).

Solution:

1.

Sample Proof. We will prove that the finite state diagram above is correct by contradiction. For a word to be accepted, it must reach the accept state \( q_2 \). For the sake of contradiction, suppose that a word that begins with a 0 and ends with a 1 is consumed, and our ending state \( q' \notin F \). From the transition function \( \delta \), we know that this state is not \( q_0 \) because the state begins with a 0. Then, the word will be consumed up until the last word and the DFA will either be in state \( q_1 \) or \( q_2 \). If the state is \( q_1 \), then \( \delta \) moves the DFA to \( q_2 \). If the state is \( q_2 \), then \( \delta \) keeps the DFA in \( q_2 \). In either case, we have a contradiction.

2.

Sample Proof. We will prove that this finite state machine is correct by construction. In particular, note that any input that terminates consumption in \( q_0 \) is of length 0, any input that terminates consumption in \( q_1 \) is of length 1, and any input that terminates consumption in \( q_2 \) is of length 2. As such, we accept any string that terminates in these states. Any other string is rejected.
3.

\{w_A | w_A \text{ begins with 0 and ends with 1 or the length of } w_B \text{ is at most 2} \}

**Problem 3**

The following language (under alphabet \{a, b\}) is a intersection of two simpler languages:

\[ L = \{w | w \text{ contains at least 2}a\text{'s and exactly 3}b\text{'s} \}\]

(a) Identify the simpler languages and design a DFA for each of them.

(b) Design a DFA that recognizes \(L\). Provide a simple proof of its correctness.

**Solution:**
1. The simpler languages:
   (a) \( \{ w_A | w_A \text{ contains at least } 2a's \} \)
   \begin{equation*}
   \begin{array}{cccc}
   & q_0 & a & q_1 \\
   b & & a & q_2 \\
   \end{array}
   \end{equation*}

   (b) \( \{ w_B | w_B \text{ contains exactly } 3b's \} \)
   \begin{equation*}
   \begin{array}{cccc}
   & q_0 & b & q_1 \\
   a & & a & q_2 \\
   b & & a & q_3 \\
   \end{array}
   \end{equation*}

2.
Proof:
There could be different proof strategies for this part, including:

(a) Walk through your construction of this DFA following the textbook approach of constructing a DFA of intersection of two languages. Make sure you describe clearly how you construct the states and transitions of this DFA from the DFAs of the two sublanguages, and how you chose the start state and accept states..

(b) Argue directly that this DFA accepts and only accepts strings in

\[ L = \{ w \mid w \text{ contains at least } 2a\text{'s and exactly } 3b\text{'s } \} \]

Make sure to prove for both directions "accept" and "only accept".
One way is to argue that the rows and columns keep track of number counts of a’s and b’s; any string in L then goes into accept state, and any string not in L will have incorrect row/column count and land somewhere other than the accept state.