CS 321: Assignment 5

Jared Wasinger

November 4, 2016

1. (a)
   i. Adversary picks p
   ii. I pick: $w = a^p b^{p+1}$

   iii. Adversary splits w into $w = xyz$
      - $xy = a^p$
      - $z = b^{p+1}$
      - $|xy| \leq p$
      - $|z| = p + 1 > 0$
   iv. I choose $i = 2$
   then, $xy^i z = xyyz$ has $p + |y|$ a's and p+1 b's.
   $p + 2|y| > p + 1$. Therefore the number of a's is greater than b's.
   $num(aa, w) \neq num(bbb, w)$. Thus, this language is not regular.

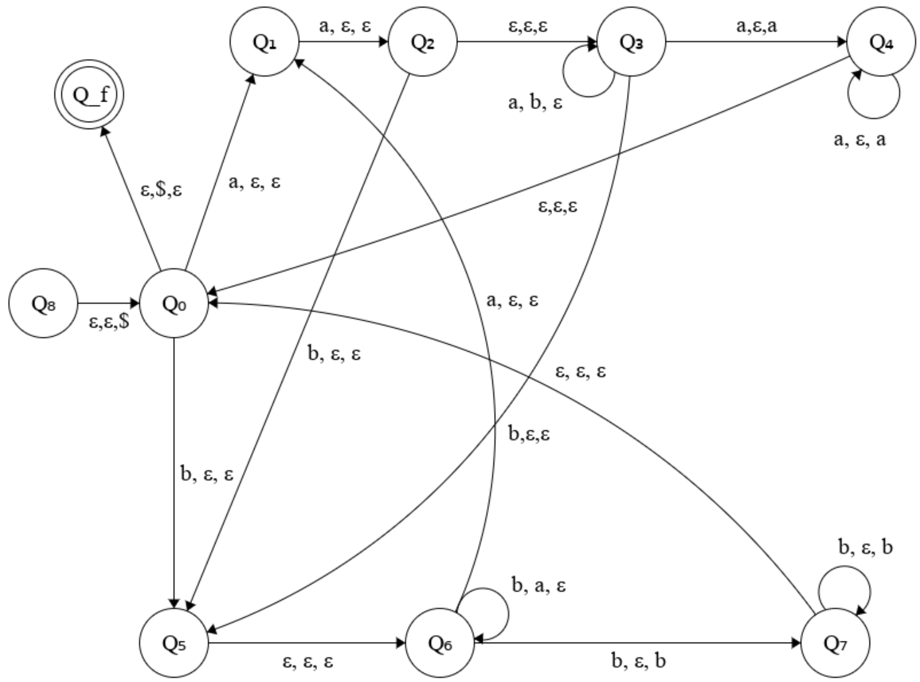
   (b)
   i. I choose $w = a^{p^2} \geq p$
   ii. adversary splits w into xyz
   $|xy| = p, |z| = p^2 - p > 0, |y| \geq p - 1$
   $|xz| = (p^2 - p) + (p - 1)$

   iii. $p^2 - 1 \neq p^2$
   iv. Proof by contradiction. The language is not regular.

2. (a) All strings where $\overline{W} = rev(W)$ follow format:
   $SubStr.ReversedSubStr$
   CFG:
   $S \rightarrow 0S1|1S0|\epsilon$

   (b) **Answer:** $S \rightarrow aSb|bSa|SS|\epsilon$

3. $\{w \in \{a, b\}^* | num(aaa, w) = num(bb, w)\}$



Explanation:

(a) The PDA starts out in state Q_8, pushing \$ onto the stack and moving to Q_0.

(b) In Q_0, the PDA can then transition to the accept state (if there are no characters to read). This is because $num(aaa, w) = num(bb, w) = 0$ at this point.

(c) The automaton reads until it has read enough consecutive a's and b's (running through Q_1, Q_2 and/or Q_5, Q_6) to have read a substring aaa or bb.

(d) When the automaton has read a desired substring (at $Q_3 or Q_6$), it will continue to loop as it reads additional instances of the same character appended to the substring.

(e) Each loop in Q_6 or Q_3 removes characters of the opposite substring type off the stack as it reads additional substrings. For example: if bb is read, the automaton advances to state Q_6 and pops and a off the stack if it is present. For each additional b read, the number of occurances of bb increases by 1. If there are additional a's present on the stack they are removed for each additional b read.

(f) When the automaton runs out of characters of the opposing substring to pop off the stack, it starts pushing characters of the substring it is reading on the stack.

(g) Thus, the end state is reached when all characters are read and the stack is empty meaning $num(aaa, w) = num(bb, w)$.

Clarification of Transitions:

(a) $Q_3 \rightarrow Q_5 = b, \epsilon, \epsilon$

(b) $Q_6 \rightarrow Q_1 = a, \epsilon, \epsilon$

(c) $Q_2 \rightarrow Q_5 = b, \epsilon, \epsilon$

(d) $Q_7 \rightarrow Q_0 = \epsilon, \epsilon, \epsilon$

(e) $Q_4 \rightarrow Q_0 = \epsilon, \epsilon, \epsilon$