CMPS 130 Homework 5

Textbook Exercises

1.29a. $A_1 = \{0^n 1^n 2^n \mid n \geq 0\}$

Demon picks p
I pick $s \in A_1 = 0^p 1^p 2^p$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = 2$

$|xy| \leq p$ implies that xy is a substring of all 0's, which implies that y is a substring of all 0's. This means that pumping y two times increases only the number of 0's.
$xy^2z = 0^{>p} 1^p 2^p \notin A_1$
$\therefore A_1$ is not regular Q.E.D.

b. $A_2 = \{www \mid w \in \{a, b\}^*\}$

Demon picks p
I pick $s \in A_2 = ab^p ab^p ab^p$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = 2$

$|xy| \leq p$ implies that xy has exactly one a, which implies that y has at most one a.

- Case 1: If y has no a's, then $xy^2z = ab^{>p} ab^p ab^p$, which cannot be in the form www.
- Case 2: If y has one a, then xy^2z has exactly four a's and cannot be in the form www.

In all cases, $xy^2z \notin A_2$
$\therefore A_2$ is not regular Q.E.D.

c. $A_3 = \left\{a^{2^n} \,\middle|\, n \geq 0\right\}$

Demon picks p
I pick $s \in A_3 = a^{2^p}$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p, |y| > 0$
I pick $i = 2$

$|xy^2z| = |xyz| + |y|$
$|xy^2z| = 2^p + |y|$ because $|s| = |xyz| = 2^p$
$|xy^2z| \leq 2^p + p$ because $|xy| \leq p \to |y| \leq p$
$|xy^2z| < 2^p + 2^p$ because $p < 2^p$
$|xy^2z| < 2^{p+1}$
Note: $|xy^2z| > 2^p$ because $|xyz| = 2^p$ and $|y| > 0$

Since $2^p < |xy^2z| < 2^{p+1}$, $xy^2z \notin A_3$
$\therefore A_3$ is not regular Q.E.D.

1.30. Example 1.73 shows that 0^p1^p cannot be pumped and still be in the language $\{0^n1^n\}$, because pumping 0^p1^p will result in $0^{>p}1^p$. However, this string is still in the language 0^*1^*, so this does not prove that 0^*1^* is irregular.

Note: This also does not prove that 0^*1^* is regular
(but we know from Kleene's theorem that any language that is described by a regular expression is regular, so 0^*1^* is regular).

Textbook Problems

1.42. Let A and B be regular languages.
The DFA that accepts $D_A = (Q_A, \Sigma, \delta_A, s_A, F_A)$ and
the DFA that accepts $D_B = (Q_B, \Sigma, \delta_B, s_B, F_B)$.

To prove that regular languages are closed under shuffle, I will construct an ε-NFA that accepts the shuffle of two regular languages A and B (using the DFAs above), which is defined as: $\{w \mid w = a_1 b_1 \ldots a_k b_k$, where $a_1 \ldots a_k \in A$ and $b_1 \ldots b_k \in B$, each $a_i, b_i \in \Sigma^*\}$

Let $N = (Q, \Sigma, \delta, s, F)$, where:
$Q = Q_A \times Q_B \times \{A, B\}$
Note: it is assumed that A and B share the same alphabet in the instructions (assuming otherwise would only make the NFA slightly more complex).
Let $x \in \Sigma$ be an arbitrary letter in the alphabet.
Then, the transition function is as follows:
$\delta\big((q_A, q_B, A), x\big) = (\delta_A(q_A, x), q_B, B)$
$\delta\big((q_A, q_B, B), x\big) = (q_A, \delta_B(q_B, x), A)$
$\delta\big((q_A, q_B, A), \varepsilon\big) = (q_A, q_B, B)$
$\delta\big((q_A, q_B, B), \varepsilon\big) = (q_A, q_B, A)$
$s = (s_A, s_B, A)$
$F = F_A \times F_B \times \{A\}$

The underlying construction of this ε-NFA starts off with the DFA constructed for the perfect shuffle of two languages (alternating between DFAs for every letter read). However, we add in ε transitions that allow a DFA to “skip” it’s turn. (Read my homework 3 problem 1.41 for a more detail)

1.46a. $L = \{0^n 1^m 0^n \mid m, n \geq 0\}$
Demon picks p
I pick $s \in L = 0^p 1 0^p$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = 2$
$xy^2z = 0^{>p} 1 0^p \notin L$
$\therefore$ L is not regular Q.E.D.

b. $L = \{0^m 1^n \mid m \neq n\}$
Demon picks p
I pick $s \in L = 0^p 1^{p+p!}$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = \frac{p!}{|y|} + 1$
Note: $1 \leq |y| \leq p$, so p! is always divisible by $|y|$
Note: $xyz = 0^{p-|y|} 0^{|y|} 1^{p+p!}$, so
$xy^i z = 0^{p-|y|} 0^{i \cdot |y|} 1^{p+p!}$
$xy^i z = 0^{p-|y|} 0^{p!+|y|} 1^{p+p!}$
$xy^i z = 0^{p+p!} 1^{p+p!} \notin L$
$\therefore$ L is not regular Q.E.D.

Alternatively, let $L = \{0^m 1^n \mid m \neq n\}$.
Assume for the sake of contradiction that L is regular. Since regular languages are closed under the complement and intersection operations, then $\bar{L} \cap 0^* 1^* = \{0^k 1^k \mid k \geq 0\}$ must be regular. However, as shown in example 1.73 (page 80 of the textbook), $\{0^k 1^k \mid k \geq 0\}$ is not regular. Therefore, our assumption must be wrong and L is not regular. Q.E.D.

c. $L = \{w \mid w \in \{0,1\}^* \text{ is not a palindrome}\}$
$\bar{L} = \{w \mid w \in \{0,1\}^* \text{ is a palindrome}\}$

Note: the following demon adversary game for $\bar{L}$ is the same as the one shown in problem 1.46a.

Demon picks p
I pick $s \in \bar{L} = 0^p10^p$ s.t. $|s| \geq p$
Demon picks $s = xyz$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = 2$
$xy^2z = 0^{>p}10^p \notin \bar{L}$
$\therefore$ $\bar{L}$ is not regular

Assume for the sake of contradiction that L is a regular language. Since regular languages are closed under the complement operation, $\bar{L}$ must be regular. However, $\bar{L}$ is not regular as shown above. Therefore, our initial assumption was wrong and L is not regular. Q.E.D.

d. $L = \{wtw \mid w, t \in \{0,1\}^+\}$
Note: the following demon adversary game cannot be the same as the one shown in problem 1.46a because $w \neq \varepsilon$.

Demon picks p
I pick $s \in L = 0^{p+1}10^{p+1}$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = 2$
$xy^2z = 0^{>p+1}10^{p+1} \notin L\}$
$\therefore$ L is not regular Q.E.D.

1.47. Let $\Sigma = \{1, \#\}$
Let $Y = \{w \mid w = x_1\#x_2\# \ldots \#x_k$ for $k \geq 0$, each $x_i \in 1^*$, and $x_i \neq x_j$ for $i \neq j\}$

Note: this proof is similar to the one I showed in 1.46b:

Demon picks p
I pick $s \in Y = 1^{p}\#1^{p+p!}$ s.t. $|s| \geq p$
Demon picks $xyz = s$ s.t. $|xy| \leq p$ and $|y| > 0$
I pick $i = \frac{p!}{|y|} + 1$
Note: $1 \leq |y| \leq p$, so p! is always divisible by $|y|$
Note: $xyz = 1^{p-|y|}1^{|y|}\#1^{p+p!}$, so
$xyz = 1^{p-|y|}1^{i \cdot |y|}\#1^{p+p!}$
$xyz = 1^{p-|y|}1^{p!+|y|}\#1^{p+p!}$
$xyz = 1^{p+p!}\#1^{p+p!} \notin Y$
$\therefore$ Y is not regular Q.E.D.

1.55e. $(01)^*$
$p_{min} = 2$
The minimum pumping length is 2.
Any string in $(01)^*$ of length 2 or greater contains a 01 group that can be pumped. However, a pumping length of 1 does not work because that would mean pumping a single symbol, which would result in a string not in the language.

f. ε
$p_{min} = 1$
By definition, the minimum pumping length of a finite language is the length of the longest string in the language plus one.
Alternatively, the minimum pumping length cannot be 0 because ε cannot be pumped (|y| must be greater than 0 in s=xyz). It can be 1 because there are no strings of length 1 or greater in the language, so the pumping lemma condition

is satisfied vacuously.

i. 1011
   $p_{min} = 5$
   By definition, the minimum pumping length of a finite language is the length of the longest string in the language plus one.
   Alternatively, the pumping length cannot be 4 because the string 1011 cannot be pumped and remain in the language. It can be 5 because there are no strings of length 5 or greater in the language, so the pumping lemma condition is satisfied vacuously.

j. Σ^*
   $p_{min} = 1$
   The minimum pumping length is 1. Σ^* describes any string that can be formed using the letters of the alphabet. Clearly, pumping any non-empty string in this language adds only letters that are already in the string (and hence the alphabet). Thus, the pumped string is obviously still in the language.