Assignment # 5.1 (M-N)

a. \( L = \{ x#y \mid x, y \in \{0,1\}^+ \text{ and } y \text{ is the two's complement of } x \} \)

Let \( R_L \) be the right invariant equivalence class defined by M-N for \( L \). Consider the pair of equivalent classes \([010^i#] \) and \([010^j#] \), \( i \neq j \), \( i,j > 0 \). \( 010^i#110^i \in L \) but \( 010^j#110^i \notin L \). Thus, for each distinct pair, \( i,j \), \( i \neq j \), \( [010^i] \neq [010^j] \) and hence \( R_L \) has infinite index.

\( L \) is not Regular by Myhill-Nerode Theorem.
Assignment # 5.1 (PL)

a. \[ L = \{ \text{x#y | x, y} \in \{0,1\}^+ \text{ and y is the twos complement of x} \} \]

PL: Gives me \(N>0\) associated with \(L\)
Me: Choose \(w=10^N#10^N\) which is in \(L\)
PL: States \(w=xyz, |xy|\leq N, |y|>0, xy^iz \in L \text{ for all } i\geq 0\)
Me: Choose \(i = 0\).
Case 1: If the initial 1 is erased, we have \(0^{N-|y|-1}#10^N\) which is not in \(L\).
Case 2: If only 0’s are affected, we have \(10^{N-|y|}#10^N\), which is not in \(L\) when \(|y|>0\) (would need to end with \(#10^{N-|y|}\)).

Thus, \(L\) is not Regular by Pumping Lemma for Regular Languages.
Assignment # 5.1b (M-N)

b. \( L = \{ a^ib^jc^k \mid i>k \text{ or } j>k \text{ or } k>i \} \)

Let \( R_L \) be the right invariant equivalence relation defined for \( L \) by M-H. Consider \([a^ib^i] \) and \([a^ib^j] \) \( i \neq j \). \( a^ib^ic^i \notin L \) but \( a^ib^jc^i \in L \)

Thus, for any two distinct \( i,j, \) \( i \neq j \), \([a^ib^i] \neq [a^ib^j] \)

This is a bit different than all our other cases in that we focus on a pattern that leads to a string not in \( L \) versus one in \( L \) – that’s a bit of a flip.
b. \( L = \{ a^i b^j c^k \mid i > k \text{ or } j > k \text{ or } k > i \} \). I found it easy to do the complement of \( L \). As regular are closed under complement, if the complement of \( L \) is not Regular, then neither is \( L \). \( L^C = \{ a^i b^j c^k \mid i \geq 0 \} \cup b^+ (a+c)^+ + c^+ (a+b)^+ \). Note that the latter two parts are regular and do not overlap the first part.

Me: \( L^C \) is regular
PL: Gives me \( N > 0 \) associated with \( L^C \)
Me: Choose \( w = a^N b^N c^N \) which is in \( L^C \)
PL: States \( w = xyz, |xy| \leq N, |y| > 0, xy^i z \in L^C \) for all \( i \geq 0 \)
Me: Choose \( i = 0 \). This says that \( xz = a^{N-|y|} b^N c^N \in L \). But, since \( |y| > 0 \) and there are no order problems (b’s before a’s or c’s, or c’s before a’s or b’s), this string is not in \( L^C \).

Thus, \( L^C \) is not Regular by Pumping Lemma for Regular Languages and, by closure of Regular under complement, \( L \) is not Regular.
c. \( L = \{ xw x | x, w \in \{a,b\}^* \text{ and } |x| = |w| \} \)

I attack this with M-N. Let \( R_L \) be the right invariant equivalence relation defined for \( L \) by M-H.

Consider \([a^iba^{i+1}]\) and \([a^iba^{j+1}]\) \( i < j \).

\( a^iba^{i+1}a^ib \in L \) but \( a^iba^{i+1}a^ib \notin L \), when \( j < i \), as last part ending \( b \) means that first part must also end in \( b \) and so each part ending in \( b \) must be preceded by \( i \) \( a \)'s. The problem then is that the middle part (the \( w \) part) would have length \( j+1 \) which is longer than the start and end parts (the \( x \) parts) which are of length \( i+1 \).

Thus, for any two distinct \( i,j \), \( i \neq j \), \([a^iba^{i+1}] \neq [a^iba^{j+1}]\). Note that, if \( i < j \), we have from above; if \( i > j \), then reverse roles of \( i \) and \( j \) and get result from above.

\( L \) is not Regular by Myhill-Nerode Theorem.
c. \( L = \{ x\,w\,x \mid x, w \in \{a,b\}^+ \text{ and } |x| = |w| \} \)

**PL:** Gives me \( N > 0 \) associated with \( L \)
**Me:** Choose \( v = a^N ba^N ba^N b \) which is in \( L \)
**PL:** States \( v = xyz, |xy| \leq N, |y| > 0, xy^iz \in L \) for all \( i \geq 0 \)
**Me:** Choose \( i = 0. \) \( a^{N-|y|} ba^N ba^N b \notin L \) since third \( b \) must have same number of \( a \)'s preceding it as first \( b \), so middle part is \( a^N ba^{|y|} \), which is longer that the starting part.

Thus, \( L \) is not Regular by Pumping Lemma for Regular Languages.
Assignment # 5.2

2. Write a regular (right linear) grammar that generates \( L = \{ w \mid w \in \{0,1\}^* \text{ and } w \text{ interpreted as a binary number has a remainder of 3 or 4 when divided by 6 } \} \).

\[
\begin{align*}
<0> & \to 0 <0> \mid 1 <1> \\
<1> & \to 0 <2> \mid 1 <3> \\
<2> & \to 0 <4> \mid 1 <5> \\
<3> & \to 0 <0> \mid 1 <1> \mid \lambda \\
<4> & \to 0 <2> \mid 1 <3> \mid \lambda \\
<5> & \to 0 <4> \mid 1 <5>
\end{align*}
\]
2. Present a Mealy Model finite state machine that reads an input $x \in \{0, 1\}^*$ and produces the binary number that represents the result of adding binary $101$ to $x$ (assumes all numbers are positive, including results). Assume that $x$ is read starting with its least significant digit.

Examples:
- $0010 \rightarrow 0111$
- $0101 \rightarrow 1010$
- $0001 \rightarrow 0110$
- $0111 \rightarrow 1100$