Question 1

Wednesday, November 13, 2024 4:33 PM

Q1 Devise a PDA that accepts the following language:

$$L = \left\{ w \in \{0,1\}^* : \frac{3}{5} \le \frac{\#_0(w)}{\#_1(w)} \le \frac{5}{8} \right\}$$

Here, $\#_x(w)$ denotes the number of occurrences of symbol x in string w.

The ratio of $\#_0$'s to $\#_1$'s must be between 3/5 inclusive and 5/8 inclusive

Derivation:

$\frac{3}{5} \le \frac{\#_0(w)}{\#_1(w)} \le \frac{5}{8}$

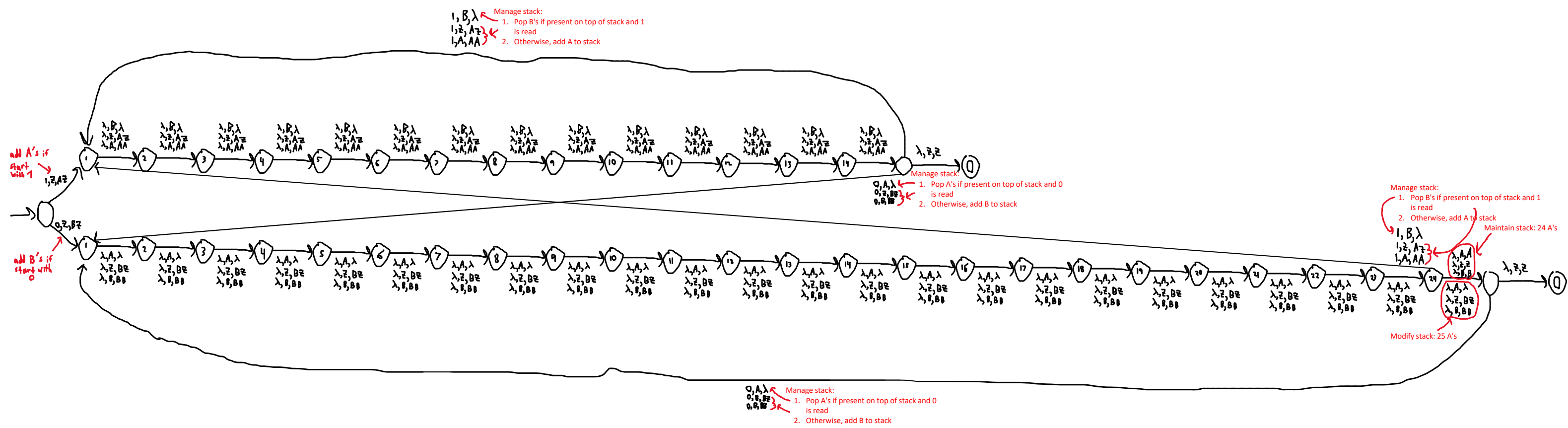
$\Rightarrow \frac{5}{3} \ge \frac{\#_1(w)}{\#_0(w)} \ge \frac{8}{5}$ (reciprocate)

$\Rightarrow \frac{8}{5} \le \frac{\#_1(w)}{\#_0(w)} \le \frac{5}{3}$ (rewrite)

$\Rightarrow \frac{8\#_0(w)}{5} \le \#_1(w) \le \frac{5\#_0(w)}{3}$ (multiply by $\#_0(w)$)

$\Rightarrow 24\#_1(w) \le 15\#_0(w) \le 25\#_1(w)$ (multiply by 15)

Managing the stack below means using A's (for 1) and B's (for 0) as a counter to track the ratio, there's many states to ensure the ratio is maintained



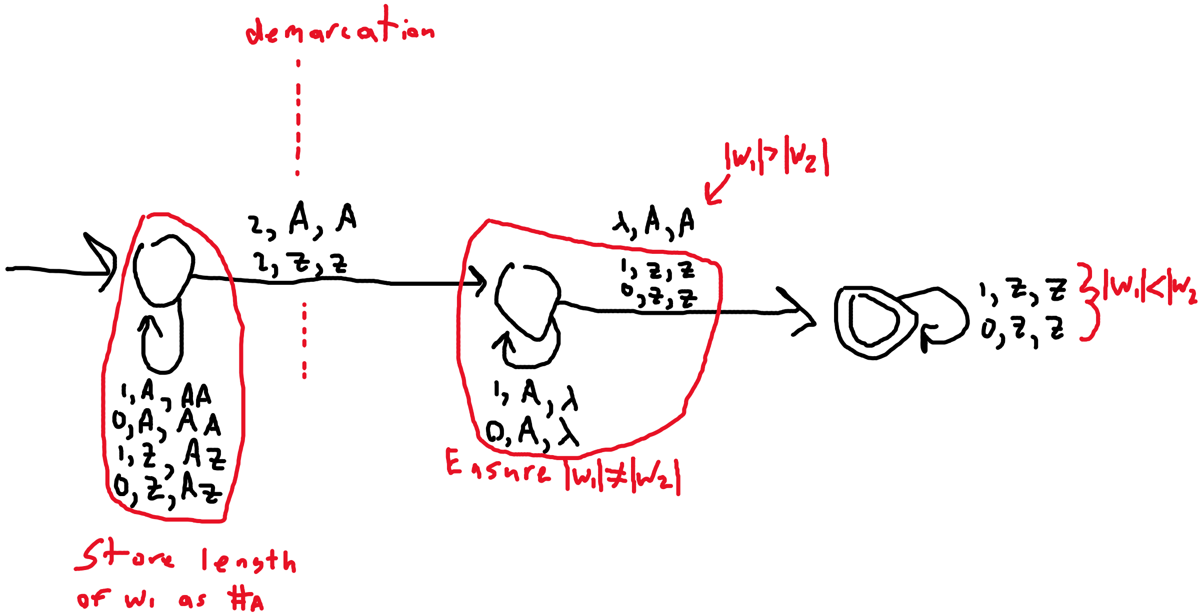
Question 2

Wednesday, November 20, 2024 5:48 PM

Q2 Devise a PDA for the following language:

$$L = \{w_1 2 w_2 : w_1, w_2 \in \{0,1\}^* \text{ and } |w_1| \neq |w_2|\}.$$

Note that the symbol 2 demarcates where w_1 ends and w_2 begins, and $|\cdot|$ denotes the length of a string.



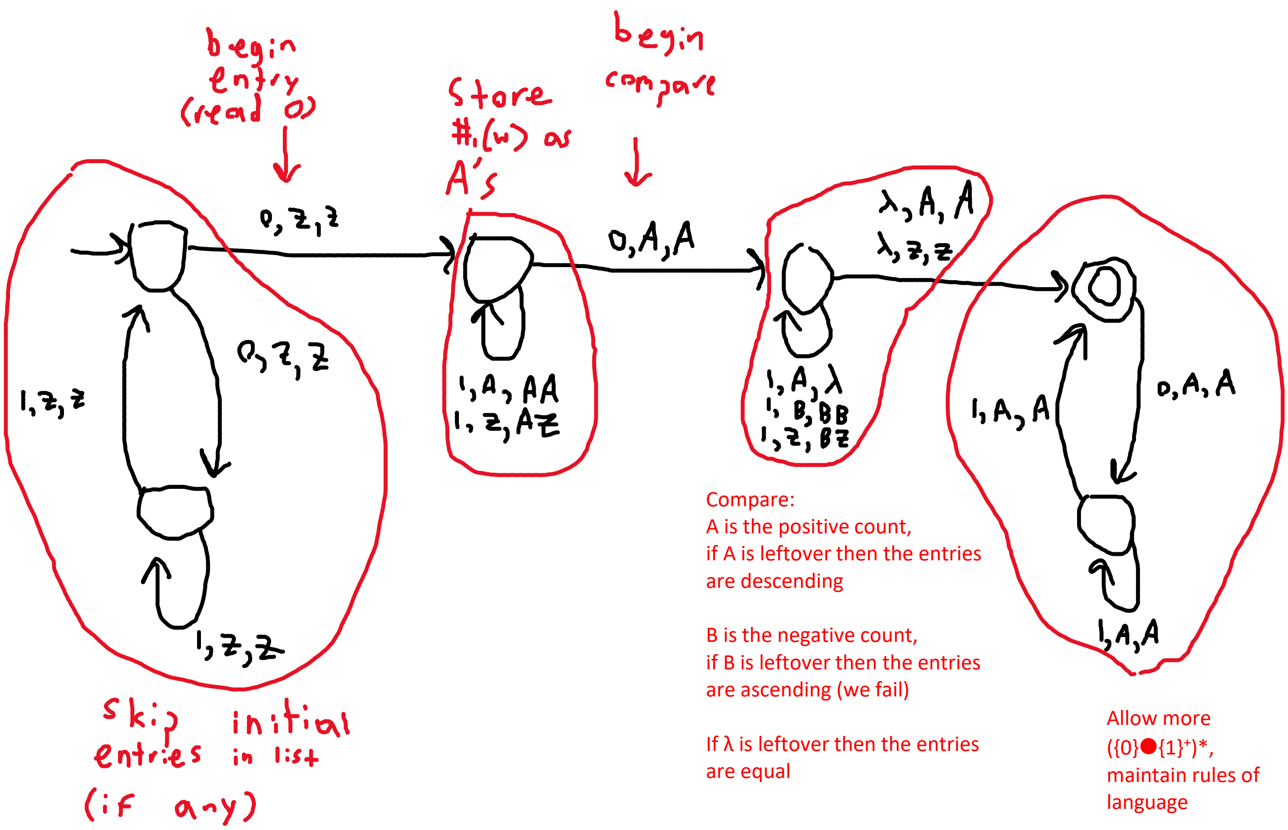
Question 3

Wednesday, November 20, 2024 5:58 PM

Q3 As in the previous assignment, let us interpret strings in $\left(\{0\} \circ \{1\}^+\right)^*$ as lists of positive numbers. For example, $(2, 1, 3)$ and $(1, 2, 3, 2)$ are represented by $01^2 0101^3$ and $0101^2 01^3 01^2$, respectively. Devise a PDA for the following language:

$$L = \left\{ w \in \left(\{0\} \circ \{1\}^+\right)^* : \text{ the entries of the list represented by } w \text{ are not in ascending order} \right\}$$

Skip initial entries, then store 1's as A's on the stack, compare using A's as a counter and insert B if we see Z to indicate descending order and to fail (since B has no future), then continue to the final state but permit future skips



Compare:
A is the positive count,
if A is leftover then the entries are descending

B is the negative count,
if B is leftover then the entries are ascending (we fail)

If λ is leftover then the entries are equal

Allow more ({0}●{1}⁺)*, maintain rules of language

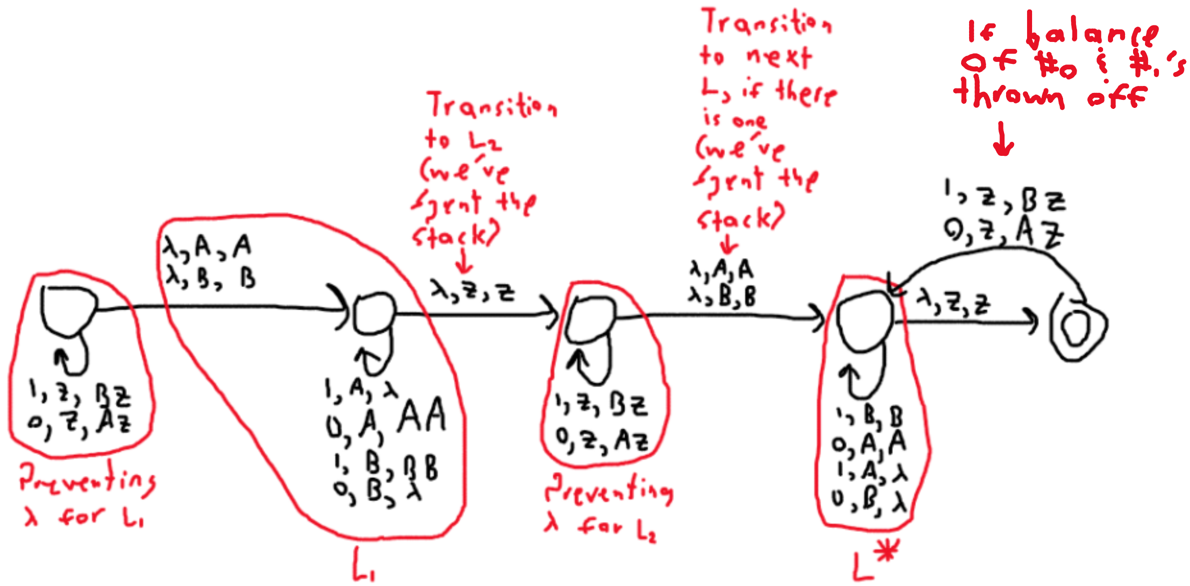
Question 4

Wednesday, November 20, 2024 7:26 PM

Q4 Let L denote the set of all **non-empty** binary strings that have equal number of zeros and ones. Devise a PDA that accepts $L \circ L \circ L^*$.

Initially for L:
Use A to keep track of $\#_0$'s
Use B to keep track of $\#_1$'s

Notation: If L●L●L*, then let the first L be L_1 and the second L be L_2 such that L_1●L_2●L*



Question 5

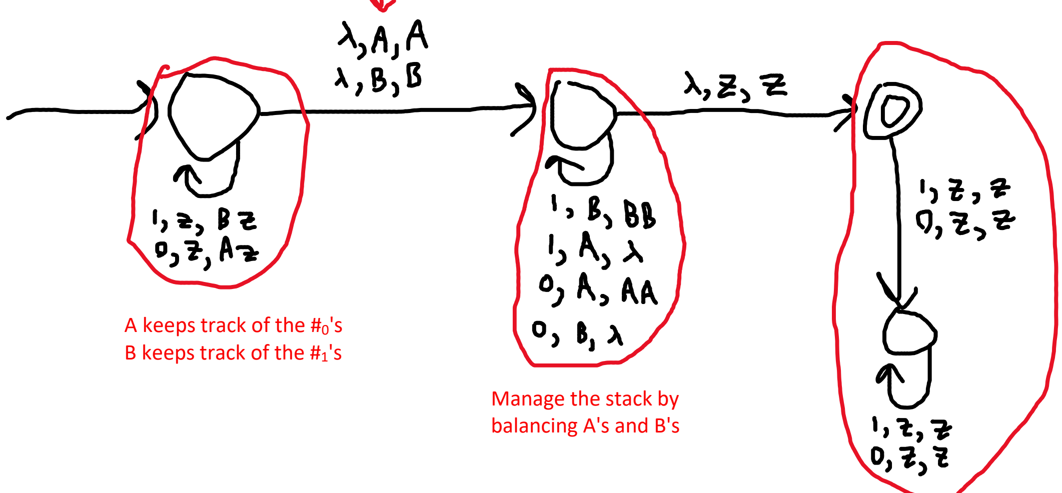
Wednesday, November 20, 2024 7:41 PM

Q5 Let L denote the folllowing language:

$$L = \left\{ w \in \{0,1\}^* : \begin{array}{c} \#_0(w) = \#_1(w) \\ \text{no non-empty proper prefix } s \text{ of } w \text{ satisfies } \#_0(s) = \#_1(s) \end{array} \right\}$$

Devise a PDA for L.

Accounts for if we had a 0 (A) or a 1 (B) previously on the top of the stack



A keeps track of the $\#_0$'s
B keeps track of the $\#_1$'s

Manage the stack by balancing A's and B's

If we had an accept state where $\#_0(w) = \#_1(w)$ (0's and 1's balanced) and we add additional 0's or 1's, then we unbalanced an already balanced proper prefix which is NOT allowed, entering a trap state

Question 6

Thursday, November 21, 2024 4:24 PM

Q6 Devise a PDA for $L_1 \cap L_2$, where

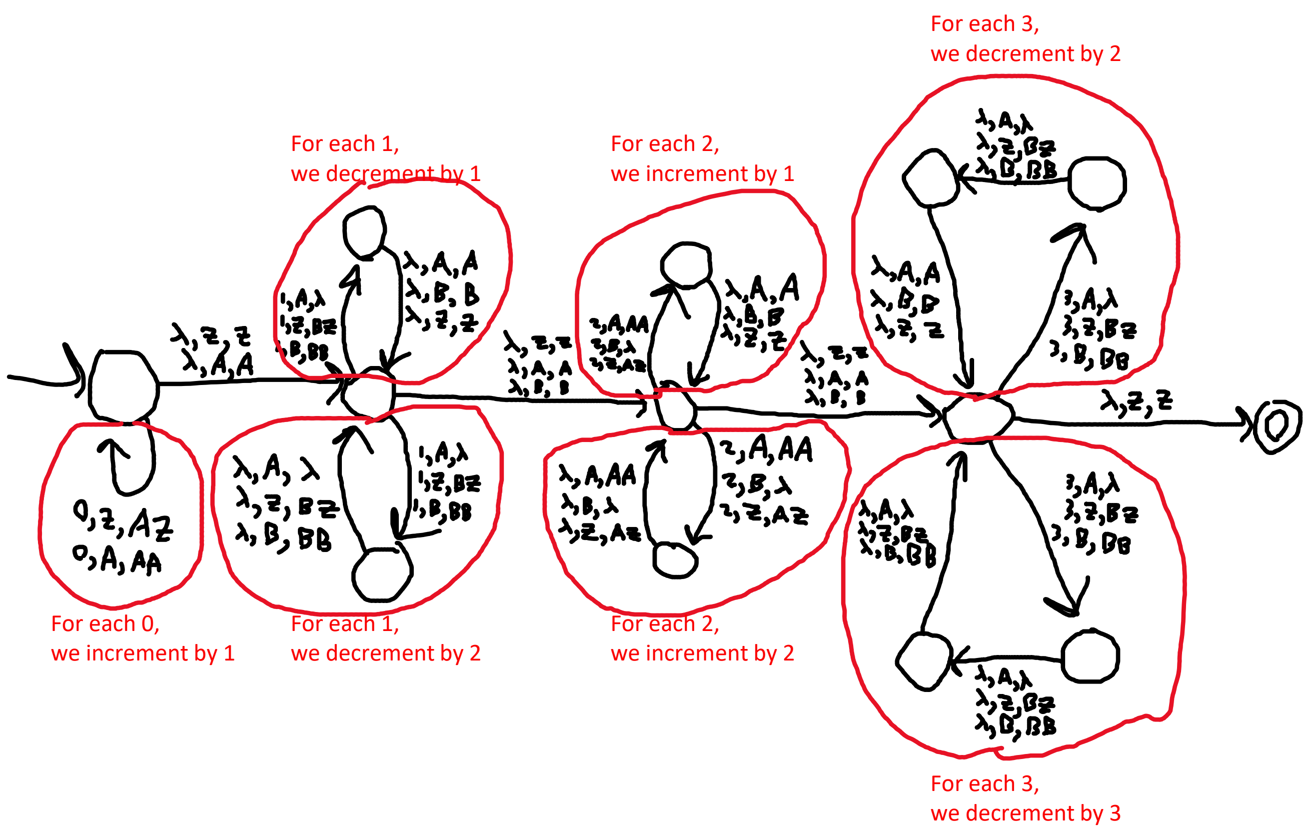
$$L_1 = \{ w \in \{0\}^* \circ \{1\}^* \circ \{2\}^* \circ \{3\}^* : \ \#_0(w) - 1\#_1(w) + 2\#_2(w) \geq 2\#_3(w) \ \}$$
$$L_2 = \{ w \in \{0\}^* \circ \{1\}^* \circ \{2\}^* \circ \{3\}^* : \ \#_0(w) - 2\#_1(w) + \#_2(w) \leq 3\#_3(w) \ \}$$

Here, $\#_x(w)$ denotes the number of occurrences of symbol x in string w.

L_1
$\#_0(w) - \#_1(w) + 2\#_2(w) \geq 2\#_3(w)$
$\Rightarrow \#_0(w) - \#_1(w) + 2\#_2(w) - 2\#_3(w) \geq 0$

L_2
$\#_0(w) - 2\#_1(w) + \#_2(w) \leq 3\#_3(w)$
$\Rightarrow \#_0(w) - 2\#_1(w) + \#_2(w) - 3\#_3(w) \leq 0$

Therefore, we can sandwich these inequalities as,
$\#_0(w) - 2\#_1(w) + \#_2(w) - 3\#_3(w) \leq 0 \leq \#_0(w) - \#_1(w) + 2\#_2(w) - 2\#_3(w)$



Question 7

Thursday, November 21, 2024 4:24 PM

Q7 As in the previous assignment andin Q3, let us interpret strings in $\left(\{0\}\circ\{1\}^+\right)^*$ as lists of positive numbers. For example, $(2,1,3)$ and $(1,2,3,2)$ are represented by 01^20101^3 and $0101^201^301^2$, respectively. Devise a PDA for the following language:

$$L = \left\{ w \in \left(\{0\}\circ\{1\}^+\right)^* : \begin{array}{l}\text{The largest entry in the list represented by } w \text{ is at least} \\ \text{twice the smallest in entry in the list represented by } w\end{array} \right\}$$

MaxEntry >= 2 * MinEntry

Two branches:
MaxEntry occurs first
MinEntry occurs first