Thomas DeMasse - HW3

1.)

State	Stack	
$q_1 \to q_2$	$[\$]$	either $2x$ 1
$q_2 \to q_2$	$\#\#\#\$$	$3x$ 1
$q_3 \to q_3$	$\#\#\$$	Rem $\emptyset$
$q_3 \to q_3$	$\$$	
$q_3 \to q_4$	$]$	
q_4		

The language recognized by this PDA is as follows: In order for the language to work, it must have either twice or triple the amount of 1s as it does 0s. For ex.) 001111 would work where 00111 would not

2.) Seperate A into 2 Separate languages $\Rightarrow S \to S_1 | S_2$

$A_1 = \{a^i b^j c^k | i, j, k \geq 0, i = j$

$\Rightarrow$ in this language we must have the Same number of a's and b's given the Condition $i = j$ for $a^i b^j$

$S_1 \to S_1 c | A | \varepsilon$

$A \to aAb | \varepsilon$

$A_2 = a^i b^j c^k | i, j, k \geq 0 \; j = k$

$\Rightarrow$ In this language we must have the same number of bs as Cs given the Condition $b^j c^k$ where $j = k$

$S_2 \to aS_2 | B | \varepsilon$

$B \to bBc | \varepsilon$

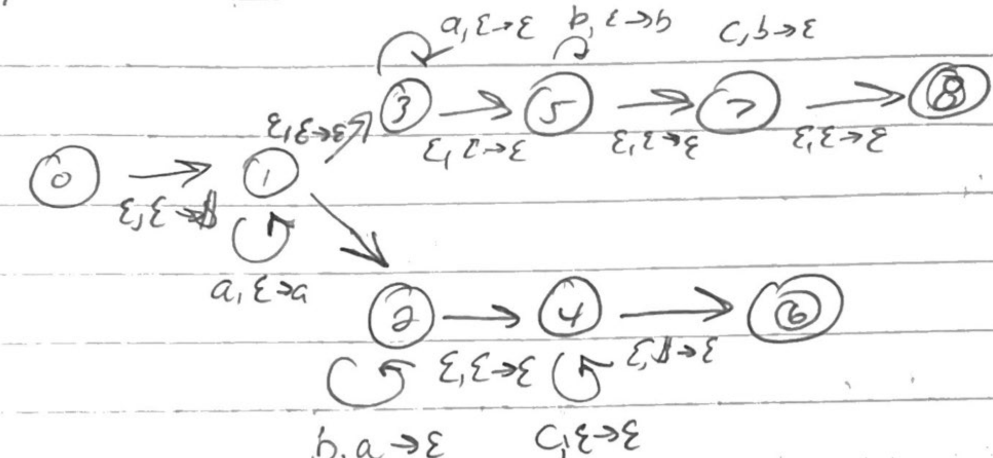
The Grammar is amigvous because each Subsiquent language derived from either S_1 or S_2 we got the "Same" result

$S \to S_1$ $\quad\quad$ $S \to S_2$

$S_1 \to A$ results in S_1 $\quad\quad$ $S_2 \to B$ results in S_2

$A \to \varepsilon$ $\quad\quad$ $B \to \varepsilon$

3.)



The PDA for the language works by reading the a's first. As it reads, when it splits off into 2 branches if either branch finds an accepting state, this causes the whole machine to acept

4.) Both language A and B are context free languages performing an intersection on the results in a new language which we hope is also context free: this being $L(A \cap B = L)$. Given the fact that L_A has b's and c's equal and L_B has a's and b's equal, the constraint on the new language must be that all a, b, c are equal

Using 2.36 from our book, it is proven that Language L is not regular. The intersection of A and B is not closed under intersection. The pumping lemma with pumping constant P can be used.

let $S = a^p b^p c^p \Rightarrow |S| \geq P$

possible vxy s.t $| \leq P$

all either a's, b's, c's and some c's and b's

are same b's and c's

Condition 1 states that: for each

$i \geq 0$ $uv^i x y^i z \in A$

however when $i = 0$ $| uv^0 x y^0 z | < |S|$

because both v and y cannot be empty,

the string will always have a character less

than that of our pumping constant P which

violates condition 3 where $|vxy| \leq P$, it's

not closed under intersection

5.) We can use the pumping lemma for CFL from

the book for simplicity sake I will use 3

as pumping constant P.

Case 1: $\{b^{3!}\} = \{bbbbbb\}$ let $i = \emptyset$

$\{uv^i x y^i z\} \rightarrow \{bb^0 bb^0 bb\}$

$= \{bbbb\} \notin L$

Case 2: $\{b^{3!}\} = \{\underbrace{bbb}_{uvx}\underbrace{bb}_{y}\underbrace{b}_{z}\}$ $i = 0$

$\{(bbb)^0 bbb\} = \{bbb\} \notin L$

Case 3: $\{b^{3!}\} = \{bbbbbb\}$ $i = 0$

$\{b^0 bbbbb\} = \{bbbbb\} \notin L$

In any case such that $P = 3$ and $i = \emptyset$, there will not exist a case that will generate a string that is an element of L because it is a factorial expression such that $b \geq 0!$ 1, 1, 2, 6, 24, 120 ... b^i there is no possible way to arrange $uvxyz$ to produce a string of $P+1$. The cases prove this where b^P will always be less than $b^{(P+1)}$