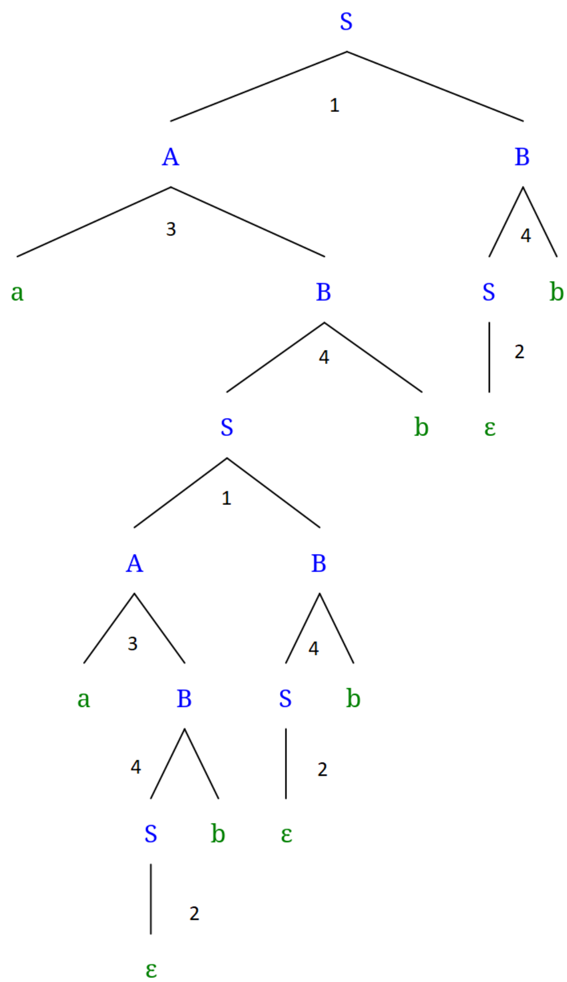
Bhagawat Chapagain - I pledge my honor that I have abided by the Stevens Honor System.



(1)

$$S \Rightarrow^1 AB \Rightarrow^3 aBSb \Rightarrow^4 aSb\epsilon b \Rightarrow^1 aABb\epsilon b \Rightarrow^{3,4}\Rightarrow aaBSbb\epsilon b \Rightarrow^4 aaSbSbb\epsilon b \Rightarrow^2 aa\epsilon b\epsilon bb\epsilon b = aabbbb$$

(2)

$S \to aaB \to aaAa$

case 1: $aaAa \to aa\epsilon a \to aaa$, however this does not equal the requested string.

case 2: $aaAa \to aabBba$

The only possibility for the B variable is Aa. So we get $aabAaba$.

case 1: $aabAaba \to aab\epsilon aba \to aababa$, however this does not equal the requested string.

case 2: $aabAaba \to aabbBbaba \to aabbAababa$

Here, considering the epsilon case for A, we find $aabbababa$ which does not equal the requested string. The other case is two b's that sandwich an Aa which means the derived string can only become a series of a's with left and right adjacent b's. Considering the original string has two consecutive b's before the penultimate and penultimate character. The grammar G cannot derive string $aabbabba$ since the deriving S derives aaB and the B variable guarantees at least one a in between any two b's if there exist any in the string.

(3) The maximum length possible is $|s| \leq k^h$ since each of the symbols generates k more symbols for each level on the tree. Solving for $h \leq \log_k(|s|)$. At each step of length l increases by $k-1$ repeatedly. Starting at length 1, we get the following expression after h steps. $|s| \leq 1 + h(k-1)$. Solving for h, we find that it is $h \geq \frac{|s|-1}{k-1}$ and also $\frac{|s|-1}{k-1} \leq h \leq \log_k(|s|)$

(4)

$X \to aXa|aYa$

$Y \to bY|\epsilon$

Remove epsilons and introduce dummy S variable. This will add the string aa to to X since Y is replaced with ϵ. We'll eliminate X and replace with its production.

$S \to X$

$X \to aXa|aYa$

$Y \to bY|\epsilon$

After the last step, we just need to introduce new variables that satisfy the restrictions of Chomsky normal form. Those being some variable $A \to BC|a$ where B, C are both variables or a terminal a but never a combination of the two.

$S \to AXA|AYA|AA$

$X \to AXA|AYA|AA$

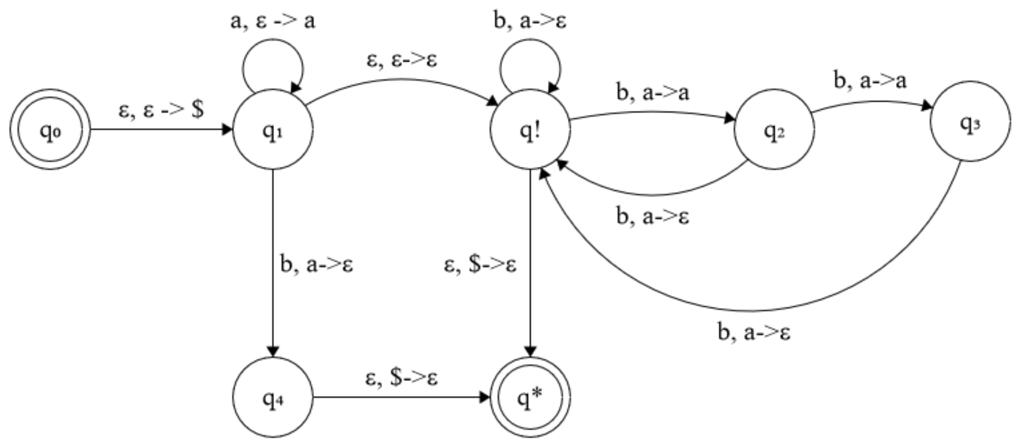
$Y \to BY$

$A \to a$

$B \to b$

(5)

The below machine has two choices from after setting up the bottom of the stack with the character \$ and adding a characters to the stack, it could see a b with an a at the top of the stack immediately then finish for the string ab or see ϵ at and attach m number of b characters where $n \leq m \leq 3n$ and n is the number of a characters. Observe that the number of b's can be any of the following $n, n+1, \ldots 2n, 2n+1, \ldots 3n$. This is why I use $q_2$and $q_3$which will detect $m = 2n$ and $m = 3n$, respectively. $q!$ serves to append the a's in between, i.e $I = \{n+1, n+2, \ldots, 2n-1, 2n+1, , \ldots 3n-1\}$. It does this by only removing a's from the stack if it has been through $2n$ or $3n$ characters and then uses the set I to fill in if the number of $b's$ is a case in-between.


q0
ε, ε -> $
q1
a, ε -> a
ε, ε->ε
q!
b, a->ε
b, a->a
q2
b, a->a
q3
b, a->ε
b, a->ε
b, a->ε
ε, $->ε
q4
ε, $->ε
q*