

Jay Patel  
HW5  
CS 331  
Professor Oliver

Q1.

As we know that  $w$  is accepted by  $L(M)$ . But in order to show the construction for  $G$ , we can tell that there is  $w$  that belongs to  $L(G)$  so  $w \in L(G)$ ; using construction proof:

For  $G$ 's construction:  $(V, \Sigma, R, S)$

So, for  $V$ :  $\{V_i \mid q_i \in Q\}$

$R$ :  $\{V_i \rightarrow aV_j \mid \delta(q_i, a) = q_j\} \cup \{V_i \rightarrow \epsilon \mid q_i \in F\}$

$S = V_0$  where  $q_0$  is the start state for DFA  $M$

So, we know that  $w$  is accepted by  $L(M)$ ,  $s_n \in F$ . There will be transition in the path for the computation, that will give us,  $s_i \rightarrow s_j$ , which exists in our grammar. Same for the states that are accepting path that is  $V_n \rightarrow \epsilon$

Therefore, in order to get to  $s_n$  you will need to follow the accept states and the rules. As we did for above, we do the same for non-accepting states.

Which then concludes these two,

$L(M) = L(G)$  for any DFA  $(M)$  and  $L(M)$  will accept and reject the same string as  $L(G)$

In other words, as the TA explained

I checked if  $w$  belongs to  $L(M)$  and then that goes and check for  $w$  belongs to  $L(G)$

Which then checks for  $c_0 \dots c_k$  belongs to  $F$

For the second case: we check for  $w$  belongs to  $L(G)$  and then we go to  $w$  belongs to  $L(M)$  which then follows to  $S \rightarrow W$  with  $S = V_1 \rightarrow W_k \rightarrow w$

Q2.

Grammar:

$S = u|v$

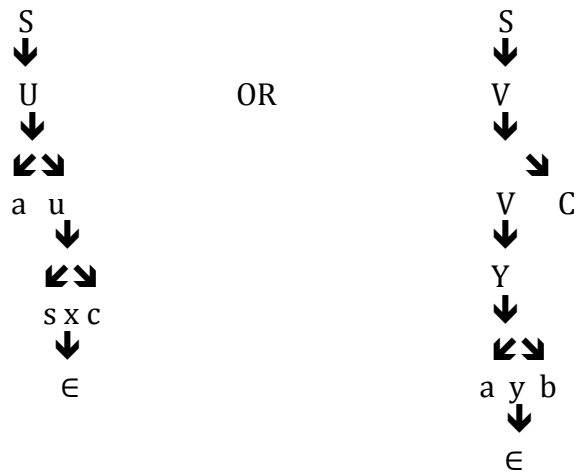
$U = au|x$

$V = vc|y$

$X = bXc|\epsilon$

$Y = aYb|\epsilon$

Context free grammar is ambiguous because strings 'abc' can draw with 2 different parse trees.



Q3.

a.  $L = \{a^n * b^m : n \neq m\} \cup \{(a \cup b)^*ba (a \cup b)^*\}$

Lets say L1 to be our Left most

And L2 to be our Right most

For L1:

$$S1 \rightarrow as|b|T|u$$

$$T \rightarrow aT|a$$

$$U \rightarrow ub|b$$

For L2:

$$S2 \rightarrow RbaR$$

$$R \rightarrow RR|a|b|^2$$

So,  $L1 \cup L2$

$$S \rightarrow S1|S2$$

$$S1 \rightarrow as|b|T|u$$

$$S2 \rightarrow RbaR$$

$$T \rightarrow aT|a$$

$$U \rightarrow ub|b$$

$$R \rightarrow RR|a|b|^2$$

The context free grammar of L is:

$$S \rightarrow TR$$

$$T \rightarrow 0T0 | 1T1 | \#R$$

$$R \rightarrow RR|0|T|E$$

b. The context free grammar is in the following,

$$P \rightarrow A|T\#A\#T|T\#A|A\#T$$

$$A \rightarrow aAb|bAa|\#|\#T\#$$

$$T \rightarrow aT|bT|\#T|\epsilon$$

Where;

A = the matched strings

T = non-terminal that as terminal of a and b.

We can tell that we can have a string of any length a and b that has can be separated by # no matter what the position is so it can be either left, right or in the middle.