COT 4210 Final Exam Part B: Context Free Grammars 5/4/2021 Solution

1) (5 pts) A context free grammar over the alphabet {a, b} with variables {S, A, B, C} and the start symbol S is given below:

S → A | BA
A → CC | AC | ε
B → CAC | aa | b
C → a | BB

Give a derivation for the string aaaabb in this grammar.

Solution
There are many answers to this question. Here is one:

S → BA
→ CACA
→ aACA
→ aACCA
→ aCCCCA
→ aaCCCA
→ aaaCCA
→ aaaaCA
→ aaaaBBA
→ aaaabBA
→ aaaabbA
→ aaaabb

Grading: 5 pts for any correct answer. Students can apply more than one rule per step as long as it's relatively easy to see what they substituted. 1 pt off per error cap at 5 pts off.

2) (7 pts) Design a PDA for the following language L, over the alphabet {a,b,c}:

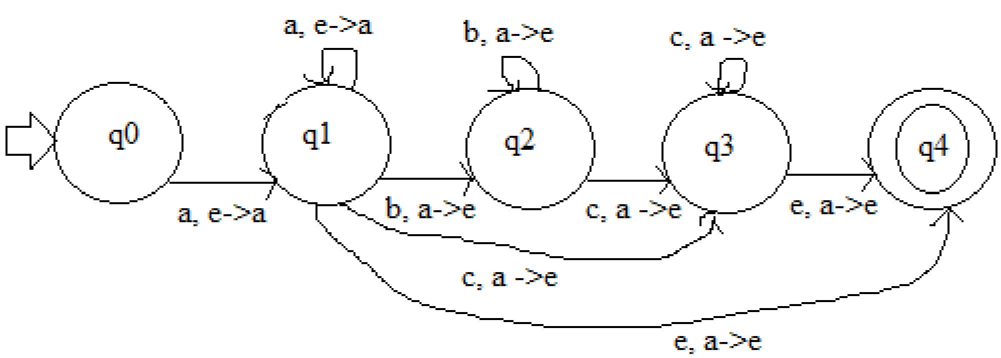
$$L = \{ a^x b^y c^z \mid x, y, z \text{ are non-negative integers such that } x > y+z. \}$$

Please give either a drawing of your design or a text description with the 6-tuple of the formal definition of the PDA. If you make a drawing, recall that a transition between states labeled as

a, b → c, means that you are reading the character a, popping character b from the stack and pushing character c onto the stack.

<u>Solution</u>

In this PDA, we read and push an 'a' before we get to the rest of our machine. (This extra state is probably unnecessary, but I already drew it so I'll just keep it in this solution.) From there, we always pop off an a from the stack for each b read in, and then c. We don't allow the letters to come out of order and we add extra transitions for either skipping b's, skipping c's or skipping both. Our very last transition reads nothing in and forces us to pop one more a off the stack ensuring that there were strictly more a's than b's and c's combined. Thus, the linear structure is what forces the letters to come in order and the use of the stack is what forces the number of a's to exceed the numbers of b's and c's combined. Here is the drawing of the PDA (note that ε is denoted as e):



Grading: 2 pts for labeling states, accept state and transitions in a valid way.
1 pt for pushing something onto the stack for each a
1 pt for popping that something off the stack when reading a b
1 pt for popping that something off the stack when reading a c
1 pt for allowing skipping b's or c's in any manner.
1 pt for popping an extra a off the stack before accepting.

3) (5 pts) In the last step of creating a Chomsky Normal Form grammar from any regular grammar, extra variables and extra rules are introduced so that no rule maps a variable to more than 2 variables. Show an application of this step on the following single rule:

$S \rightarrow ABCDEFG$

You may add additional variables that are English capital letters, H or later in the alphabet.

Solution

There are many ways to do this. The key is to use the extra variables to "daisychain" the full rule in some manner:

$S \rightarrow AI$
$I \rightarrow BJ$
$J \rightarrow CK$
$K \rightarrow DL$
$L \rightarrow EM$
$M \rightarrow FG$

Grading: 3 pts if when you chain S it creates ABCDEFG
2 pts if all the rules go to exactly 2 variables.

4) (8 pts) Let $\Sigma = \{a, b, c\}$ and $L = \{w \mid w$ has the same number of a's, b's and c's$\}$. Prove that L is not context free via the Pumping Lemma for Context Free Grammars.

Solution

To prove that L is not Context Free, we must find a string of length p or greater that belongs to L, which doesn't satisfy the Pumping Lemma for Context Free Grammars. Let p be the pumping length of L. Consider the string $s = a^p b^p c^p$. This string is in L since it has the same number of a's b's and c's. The pumping lemma states that there must be a way to split of the string such that s = uvxyz such that for all non-negative integers i, uv^ixy^iz is in L, $|vy| > 0$ and $|vxy| \le p$. The string vxy must fit in one of these categories based on these restrictions, no matter how we subdivide it:

1) some positive number of a's
2) some positive number of a's followed by a positive number of b's
3) some positive number of b's
4) some positive number of b's followed by a positive number of c's
5) some positive number of c's

Note: it can't have a's b's and c's because there is no substring of s with length p or less with all 3 distinct characters.

In all of these five cases, consider the string uv^2xy^2z. This string must have exactly copies of 1 or 2 distinct letters that have been added compared to s. This means that this string will have exactly p of one of the three characters, and definitely more than p of another of the three characters, proving that this resultant string is NOT in L, no matter how u, v, x, y and z are selected. (For example, in case 1, the string would have more a's than b's. In case 2, the string would have more a's than c's. In case 3, the string would have more b's than c's. In case 4, the string would have more b's than a's. In case 5, the string would have more c's than a's.) Thus, L does not satisfy the Pumping Lemma for Context Free Grammars. We can conclude that L is NOT Context Free.

Grading: 3 pts for picking a string
3 pts for considering all ways to split the string
2 pts for finding a string of the appropriate form not in the language and proving that it is not.