ECE 468 & 573
Problem Set 2: Context-free Grammars, Parsers

1. For the following sub-problems, consider the following context-free grammar:

$$
\begin{aligned}
S &\rightarrow A && (1)\\
A &\rightarrow xAC && (2)\\
A &\rightarrow B && (3)\\
B &\rightarrow yBC && (4)\\
B &\rightarrow \lambda && (5)\\
C &\rightarrow z && (6)
\end{aligned}
$$

   (a) What are the terminals and non-terminals of this language?

   (b) Describe the strings are generated by this language. Is this a regular language (*i.e.*, could you write a regular expression that generates this language)?

   (c) Show the derivation of the string $xxyzzz$ starting from S (specify which production you used at each step), and give the parse tree according to that derivation.

   (d) Give the first and follow sets for each of the non-terminals of the grammar.

   (e) What are the predict sets for each production?

   (f) Give the parse table for the grammar. Is this an LL(1) grammar? Why or why not?

   (g) Add one more production for C (i.e., of the form $C \rightarrow \alpha$) that makes this grammar *not* LL(1).

2. for the following sub-problems, consider the following grammar:

$$
\begin{aligned}
S &\rightarrow AB\$ && (7)\\
A &\rightarrow xB && (8)\\
A &\rightarrow xyB && (9)\\
B &\rightarrow zA && (10)\\
B &\rightarrow w && (11)\\
& && (12)
\end{aligned}
$$

   (a) Describe the strings generated by this language.

   (b) Is this language LL(1)? Why or why not?

   (c) Build the CFSM for this grammar.

   (d) Build the goto and action tables for this grammar. Is it an LR(0) grammar? Why or why not?

(e) If we add the production

$$B \rightarrow yz$$

to the grammar, is it an LR(0) grammar? Why or why not?

(f) (ECE 573 only): Build the LR(1) machine for the grammar extended with the rule from (e).