ECE 468
Problem Set 3: LR(0) parsers

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

$$
\begin{aligned}
S &\rightarrow E\$ && (1)\\
E &\rightarrow (E+E) && (2)\\
E &\rightarrow (E-E) && (3)\\
E &\rightarrow x && (4)\\
& && (5)
\end{aligned}
$$

   (a) Build the CFSM for this grammar. Label the states in the machine as "shift", "reduce", or "accept" states.

   (b) Show the actions the parser would take while parsing $(x + (x - x))\$$. For shift actions, just say "shift". For reduce actions, say "reduce" and indicate which production you are reducing. You do not have to indicate which state you wind up in after the reduction is complete.

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

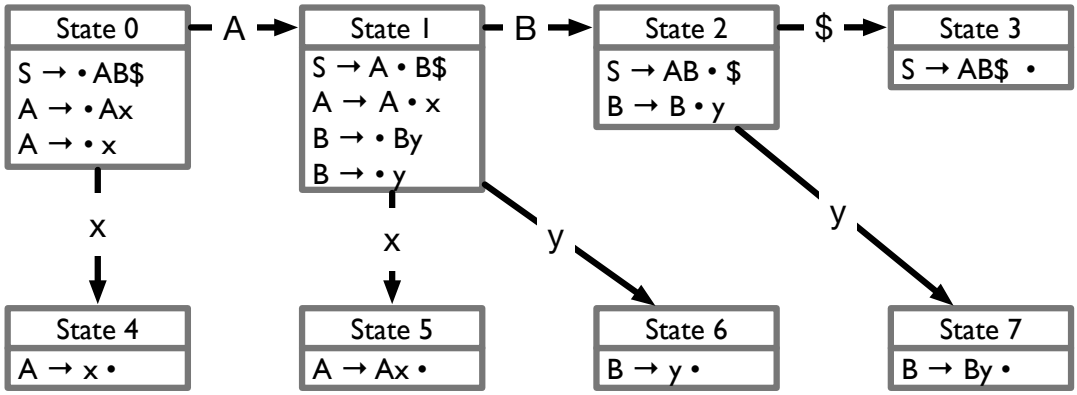
$$
\begin{aligned}
S &\rightarrow P\$ && (1)\\
P &\rightarrow [P && (2)\\
P &\rightarrow Q && (3)\\
Q &\rightarrow [Q] && (4)\\
Q &\rightarrow x && (5)
\end{aligned}
$$

   (a) Build the CFSM for this grammar.

   (b) Is this grammar an LR(0) grammar? Why or why not?

   (c) Consider the string $[x\$$. If the grammar *is* LR(0), show the actions your parser would take while parsing the string. If the grammar *is not* LR(0), explain, using this string as an example, why the parser gets stuck.

3. For the following sub-problems, consider the CFSM below:



(a) What is the grammar that this CFSM was derived from?

(b) Show the action the parser would take when parsing the string $xxyy\$$. For shift states, list the action as "Shift X" where X is the state that the parser moves to (for example, "Shift 5"). For Reduce states, list the action as "Reduce P, goto Y", where P is the production being reduced, and Y is the state the parser is in after completing the reduction (for example, "Reduce $A \rightarrow Ax$, goto 1").