CS 525: Theory of Computation
Problem Set 3

Dustin Ingram, Aaron Rosenfeld, Eric Simon

February 28, 2012

4.10 **Solution:** Here, we assume that we are already able to determine if a CFG contains a cycle (if any word has a derivation which takes 2^b derivations or more, where b is the number of variables in the CFG, it is infinite, etc.):

1) Convert M to a CFG C.
2) If C contains a production cycle, $L(M)$ is infinite

4.12 **Solution:** Here we must simply prove that $L(R) \setminus L(S) \neq \emptyset$

1) $L(R) \subseteq L(S) \Leftrightarrow L(R) \cap L(S) = L(R)$
2) Since regular languages are closed under intersection, $L(R) \cap L(S)$ is regular.
3) If $L(R) \cap L(S) = L(R)$, accept, using EQ_{DFA}.

4.22 **Solution:** For every state $\{q_1, q_2, \ldots, q_n\}$ in the PDA, create a CFG_{q_n} for the PDA for which S_{q_n} is the only accepting state. If $L(\text{CFG}_{q_n}) = \emptyset$ for any CFG_{q_n}, S_{q_n} is a **useless state**. The language would be as follows:

$$\text{USELESS}_{PDA}\{\langle P \rangle | \text{P is a PDA that has useless states}\}$$