I. (10 pts.) Design a context-free grammar for the language 
\( \{xx^Ryy^R \mid x, y \in \{a, b\}^+, abb \text{ is a substring of } x, \text{ and } |y| \text{ is odd} \} \)

\[
S \rightarrow S_1 S_2 \\
S_1 \rightarrow a S_2 a | b S_1 b | a a | b b \\
S_2 \rightarrow a S_2 a | b S_2 b \\
S_1 \rightarrow a S_1 a | b S_1 b \\
x \rightarrow a x a | b x b | \epsilon
\]

* I was not planning to write solutions to these problems. But the TA & a couple of students in the class to post the solutions to these problems.

4/15/2013
II. (10 pts.) Establish the decidability of one of the following problems by designing an appropriate algorithm.

- Given a CFG \( G \), is there an \( x \in L(G) \) such that \( |x| \) is even?
- Given \( [M_1] \) and \( [M_2] \), \( M_1 \) and \( M_2 \) being DFA language recognizers, is \( L(M_1) \cap L(M_2) \) an infinite set?

To simplify the description, we assume that the CFG is in chomsky normal form. But the proof can be extended to any form. For each nonterminal \( A \), we create \( A \rightarrow \alpha \) and we create a set \( \Delta \) of

\[
\begin{align*}
A \rightarrow \alpha & \quad \text{if } \alpha \text{ is even, and} \\
A \rightarrow \beta & \quad \text{if } \alpha \text{ is odd.}
\end{align*}
\]

Initially \( \Delta \) is empty.

For each \( A \rightarrow \alpha \), add \( A \rightarrow \alpha \) to \( \Delta \).

For each \( A \rightarrow \beta \), add \( A \rightarrow \beta \) to \( \Delta \).

Repeat until the size of \( \Delta \) does not increase

For each \( A \rightarrow BC \), and for each \( \text{ev in } \Delta \implies \text{add } A \rightarrow AB \),

\( \text{ev in } \Delta \implies \text{add } A \rightarrow CB \).

Since the max size of \( \Delta \) is \( 2N \) if \( N \) is the number of nonterminals, the process stops in a finite number of steps.

Finally check whether \( \text{ev in } \Delta \), if so output yes, else output no.

Claim: If \( M \) and \( M_2 \) have \( 2 \) and \( 2 \) states, then \( L(M_1) \cap L(M_2) \) is infinite if \( M_1 \) and \( M_2 \) accept a string \( x \) s.t. \( 3 \leq |x| \leq 23 \).

Thus, check whether a string of length in the interval \( [3, 23] \) is accepted by both \( M \) and \( M_2 \).

An additional sheet is attached.
If \( M_1, M_2 \) accept a string of length in \([5, 23]\), let one such string be \( a_1 a_2 \ldots a_L \) s.t. \( 5 \leq L \leq 23 \). Let the pairs of states reached by \( M_1, M_2 \) on this string be \((P_0, Q_0), (P_1, Q_1), \ldots, (P_k, Q_k)\). Up to \( M_1, M_2 \) and \( Q_2 \) for \( M_2 \), since there are only 3, \( M_2 \) distinct pairs, \( a_j \) the pairs are equal. Let they be \((P_i, Q_i) \equiv (P_j, Q_j) \forall \) the string. Then \( a_j \ldots a_l, a_{l+1}, \ldots, a_k \), \( k \geq 0 \), are accepted by both \( M_1, M_2 \). This is a set of infinite strings.

If \( L(M_1) \subset L(M_2) \) is an infinite set, it must contain a string of length \( \geq 38 \). Let a shortest such string be \( a_1 \ldots a_L \). If \( L \leq 23 \), we are done.

If \( L > 23 \), track the pairs as above and cut off a substring of length \( \leq 3, 13 \) resulting in a shorter string of length \( \geq 3 \) that is accepted. This violates the minimality assumption.
III. (10 pts.) Design a Turing machine for computing the following function.

\[ f(x, y, z) = \begin{cases} 
  x + (y - z) & \text{if } y \geq z \\
  x & \text{otherwise}
\end{cases} \]

I am skipping this simple problem which almost every student got it correct.
IV. (10 pts.) Solve one of the following problems.

- Prove the undecidability of the following problem. Given CFGs $G_1$ and $G_2$, is there an $x \in L(G_1) \cap (L(G_2))^R$, and $|x|$ is even? (Hint: Reduce the Post Correspondence Problem to this problem.

- Recall that the Uniform Halting Problem (UHP) asks whether a given Turing machine halts on every input. Reduce the UHP to the problem of testing whether a given TM computes the identity function; i.e., the function $f$ s.t. $f(x) = x$ for every $x$.

\[ \text{Typical } E = \langle x, y, z, i \rangle, \quad \text{and } \quad \langle x, y, z \rangle \quad \text{ind.} \]

Given $E$, we want to transform it $(G_1, G_2)$ s.t. $E$ has a solution iff $\exists x \in L(G_1) \cap (L(G_2))^R$ s.t. $|x|$ is even.

\[ G_1 \quad \text{S} \rightarrow x_1 s_1 | x_2 s_2 | x_3 s_3 | \ldots | x_n s_n | x_1 x_2 x_3 \ldots x_n \]

\[ G_2 \quad \text{S} \rightarrow y_1 s_1 | y_2 s_2 | y_3 s_3 | \ldots | y_n s_n | y_1 y_2 y_3 \ldots y_n \]

It is easily seen that $x_i = y_i \forall i \iff x_1 x_2 x_3 \ldots x_n \in L(G_1) \cap (L(G_2))^R$.

Note also that $|x_1 x_2 x_3 \ldots x_n| \in L(G_1) \cap (L(G_2))^R$ is of even length.

Note that the transformation from $E$ to $(G_1, G_2)$ is computable.

\[ \text{Typical } [M] \quad \langle M' \rangle \quad \text{ind.} \]

Transform $[M]$ to $[M']$ s.t. TM $M'$ halts on every input $x$ iff $x \in L(M)$.

TM $M'$ computes $f(x) = x$ for every $x$ on a separate pad of tape.

On any input $x$, $M'$ simulates $M$ on input $x$.

M halts, $M'$ cancels the computation & leaves $x$ on tape.

The rest is easy.