Problem Set 2

Read all of Chapters 3 and 4.

0.1 Read and solve, but do not turn in: Book, 2.16. [CFLs closed under $\cup$, $\circ$, $\ast$]
Solve by using both CFLs and PDAs.

0.2 Read and solve, but do not turn in: Book, 2.18. [CFL $\cap$ regular $=$ CFL]
You can check your solution with the one in the book.

0.3 Read and solve, but do not turn in: Book, 2.26. [Chomsky normal form]

1. Let $C = \{rst \mid r, t \in 0^* \text{ and } s \in 0^*10^* \text{ where } |r| = |s| = |t|\}$. Show that $C$ is a CFL in two ways:
   (a) by giving a CFG that generates $C$, and
   (b) by giving a PDA that recognizes $C$.

2. Let $D = \{rst \mid r, s, t \in 0^*10^* \text{ where } |r| = |s| = |t|\}$. Here $\Sigma = \{0, 1\}$.
   (a) Show that $D$ is not a CFL.
   (b) Is $D \cup (\Sigma\Sigma\Sigma)^* \text{ a CFL?}$ Prove your answer.
   (c) Is $D \cup \Sigma(\Sigma\Sigma\Sigma)^* \text{ a CFL?}$ Prove your answer.

3. Say that a variable $A$ in CFG $G$ is usable if it appears in some derivation of some string $w \in L(G)$. Given a CFG $G$ and a variable $A$, consider the problem of testing whether $A$ is usable. Formulate this problem as a language and show that it is decidable.

4. A queue automaton is like a push-down automaton except that the stack is replaced by a queue. A queue is a tape allowing symbols to be written only on the left-hand end and read only at the right-hand end. Each write operation (we’ll call it a push) adds a symbol to the left-hand end of the queue and each read operation (we’ll call it a pull) reads and removes a symbol at the right-hand end. As with a PDA, the input is placed on a separate read-only input tape, and the head on the input tape can move only from left to right. The input tape contains a cell with a blank symbol following the input, so that the end of the input can be detected. A queue automaton accepts its input by entering a special accept state at any time. Show that a language can be recognized by a deterministic queue automaton iff the language is Turing-recognizable.

5. Show that a language is decidable iff some enumerator enumerates the language in string order. (String order is the standard length-increasing, lexicographic order, see text p 14).

6. Let $C$ be a language. Prove that $C$ is Turing-recognizable iff a decidable language $D$ exists such that $C = \{x \mid \exists y \in \{0, 1\}^* ((x, y) \in D)\}$. (Hint: You must prove both directions of the “iff”. The (←) direction is easier. For the (→) direction, think of $y$ as providing additional information that allows you to confirm when $x \in C$, but without the possibility of looping.)

7. *(Optional) Recall the MS operation on languages we defined in Problem Set 1. Is the class of CFLs closed under MS? Prove your answer.