CS500, Theory of Computation
Homework #2

Note: You may discuss this homework with others in the class. However, you must do your own writeup, and you must clearly state on your homework who you worked with. Due by email in .pdf format by midnight on Sunday, February 29th.

1. Consider the two languages

$$L_1 = \{a^i b^j c^k d^\ell \mid i = j \wedge k = \ell\}$$

$$L_2 = \{a^i b^j c^k d^\ell \mid i = k \wedge j = \ell\}$$

   Show that L_1 is context-free but L_2 is not.

2. Show that both the context-free languages and the deterministic context-free languages are closed under intersection with regular languages. That is, suppose R is regular; show that if L is context-free, then so is $L \cap R$, and if L is deterministic context-free, then so is $L \cap R$.

3. Inspired by the previous problem, let L be the parenthesis language $\{\epsilon, (), (()), ()(), \ldots\}$ and R the regular language where neither '(' nor ')' can occur more than 3 times in a row. Give a context-free grammar for $L \cap R$.

4. Recall that a grammar in *Chomsky normal form* is one where all rules are of the form $A \to BC$ or $A \to a$, where capital and lower-case letters represent variables and terminals respectively. Let G be a grammar in Chomsky normal form with $|V| = k$ variables, and let L be the language generated by G. Show that if L contains a word of length greater than 2^{k-1}, then L is infinite.

5. Show that the language

$$L = \{a^i b^j c^k \mid i \neq j \vee j \neq k\}$$

   is context-free but not deterministic context-free.

6. Show that the *complement* of the set of palindromes, $L = \{w \in \{a, b\}^* \mid w \neq w^R\}$, is context-free. Since both L and $\overline{L}$ are context-free, does this mean that they are deterministic context-free? Give a DPDA that recognizes L, or some intuition for why one does not exist.

7. We saw in class that while $\{a^n b^n c^n\}$ is not context-free, it is the intersection of two context-free languages. Consider a k-symbol alphabet $\{a_1, \ldots, a_k\}$, and the language $L_k = \{a_1^n a_2^n \cdots a_k^n\}$. What is the smallest number of context-free languages such that L_k is equal to their intersection?

8. Show that the language

$$L = \{ucv \mid u, v \in \{a,b\}^*, u \neq v\}$$

is context-free. (Note that u and v are not required to have the same length, and c is used as a marker.) Hint: first, design a grammar for

$$\{ucv \mid u, v \in \{a,b\}^*, |u| = |v|, \text{ and } u \text{ and } v \text{ differ in their last symbol}\}$$

and then figure out how to extend u and v with arbitrary strings.

9. Let L be the copy language,

$$L = \{ww \mid w \in \{a,b\}^*\} \quad .$$

Show that its complement $\overline{L}$ is context-free. This is tricky, but the idea is similar to the previous problem.

10. Consider the language

$$L = \{w \in \{a.b\}^* \mid \#_a(w) = \#_b(w)\}$$

i.e., words with an equal number of as and bs.

   (a) How many words $N(\ell)$ are there of each length ℓ?

   (b) What is its generating function $g(z) = \sum_\ell N(\ell)\, z^\ell$? (Feel free to ask Mathematica or Maple to sum the series.)

   (c) Recall the unambiguous grammar for the bracket language, $S \to (S)S \mid \epsilon$. Inspired by this, we might hope that

$$S \to aSbS \mid bSaS \mid \epsilon$$

   is an unambiguous grammar for L, but unfortunately it's not. Explain why.

   (d) Now construct an unambiguous grammar for L. Hint: what kind of paths — tracking the depth of the stack, or tracking the imbalance between as and bs seen so far — do words in L correspond to, and how can we unambiguously define them as being made up of smaller paths?

   (e) Use your unambiguous grammar from (d) to derive the generating function $g(z)$ and check that it gives the same answer as you gave in (b) above.