CSE396, Spring 2026 Problem Set 5 Due Fri. 3/13, 11:59pm

Reading and Recitations:

Next week will cover proofs that context-free grammars G generate certain languages L. The text alludes to this only briefly. I will show how to *abbreviate* proofs of "$L(G) \subseteq L$" (in words: the grammar G is *sound* for the specification of L) by assigning **properties** to P_A each variable A and verifying that each rule preserves them. "Preserves" means the following: Suppose you use a rule $A \to \vec{X}$ to derive a (sub-)string w. Suppose there are k variables $B_1, \dots, B_k$ inside $\vec{X}$ that derive substrings $u_1, \dots, u_k$ of w. Here the $B_1, \dots, B_k$ need not be distinct and some might even be A itself. What you want to show is that **if** each u_i satisfies the property P_{B_i}, **then** the whole string w satisfies the property P_A. The required handout on the course webpage,

https://cse.buffalo.edu/˜regan/cse396/CSE396SI.pdf

lays out a "proof script" for abbreviating these proofs under the name "Structural Induction" (SI). The script handles what formally are multiple-threaded inductions on the number n of steps in grammar derivations in a sanitized-but-vivid manner that doesn't need mention of "n" at all. That is, it inducts directly on the structure of the grammar. The Tuesday lecture will also draw analogies to inductive thinking about object-oriented hierarchies or (for those in CSE305) recursively defined datatypes.

Proofs of "$L \subseteq L(G)$"—an idea I've called "comprehensiveness"—need to keep the "n," however, and hence can be rather painful.

This homework, however, has none of these proofs yet and is based on lectures already given before Prelim I. Except, well, part (3a) below acts as a preview of "SI." There are also some sub-surface connections between problem (2) and Q7-Q8 on the TopHat portion.

———-**Homework**—part online and all *individual work*—due **Fri. 3/13, 11:59pm**———-

(1) Using *TopHat*, the "Worksheet" titled **Spr 2026 HW5 Online Part**. There are 10 questions, each worth 2 points, for 20 total. (Recall the TopHat deadline is strict.)

(2) Let $B = \{x \in \{a,b\}^* : x \text{ has odd length with a } b \text{ in the middle}\}$. Design a CFG G_B such that $L(G_B) = B$. Define A similarly with odd length and an a in the middle and write down the analogous CFG G_A.

Now let S_B and S_A be the start variables (maybe the only variables) of your two grammars. Define a new grammar G by adding a new start symbol S and the two rules

$$S \longrightarrow S_A S_B \mid S_B S_A.$$

Does $L(G)$ contain any "double words" of the form $w = xx$ where $x \in \{a,b\}^*$? Does it contain all the even-length words that aren't double words? Give some justification for your answers. (24 pts. total, split 12 for G_A and G_B and 3+3+6 for the two questions at the end plus the justification.)

(3) Define the grammar G with one variable E and terminal alphabet $\Sigma = \{0, \$, d\}$ by the following rules:

$$E \longrightarrow \epsilon \mid 0E \mid \$E \mid \$EdE$$

The interpretation again is that a word w in Σ^* is a one-dimensional "dungeon" each of whose rooms is either empty (0), contains a spear (\$), or has a dragon (d). The rules of play now allow saving and carrying as many spears as you are lucky enough to find, though you must still leave one spear stuck in any dragon you kill. Now the dungeon $\$d\$\$dd$ (which killed the DFA implementing the rule of carrying one spear only) is survivable: you lose a spear killing the first dragon but get to save two spears to kill the last two dragons. But $\$d\$dd\$$ is still not survivable because the third spear comes too late.

(a) Explain in words why the rules of G maintain the property that every word w generated by G is a dungeon in which the player survives.

(b) Give a parse tree for $\$d\$\$d\$dd$.

(c) Show that $0\$0\$0d$ is ambiguous in G: which spear killed the dragon?

(d) Can you give a parse tree for $\$d\$\$ddd\$$? Why or why not?

($4 \times 6 = 24$ pts., for 68 total on the set.)