CS46 practice problems 10

These practice problems are an opportunity for discussion and trying many different solutions. It is **not counted towards your grade**, and **you do not have to submit your solutions.** The purpose of these problems is to get more comfortable with reasoning and writing proofs about decidability, recognizability, and co-recognizability.

If you are stumped or looking for guidance, **ask**.

1. Consider the language $L = \{\langle M, w\rangle \mid M$ is a single-tape TM that never modifies the portion of the tape that contains the original input $w\}$.

   (a) Show that L is co-Turing-recognizable, by briefly describing the elements of $\overline{L}$ and then describing a recognizer for $\overline{L}$.

   (b) Is L decidable? Prove your answer.

   Note that if you can show that L is Turing-recognizable, then you can apply Theorem 4.22 and part (a) to show L is decidable.

2. For each of the following languages, review if the language is decidable, Turing-recognizable, co-Turing-recognizable, or none of these. A_{DFA}, A_{CFG}, A_{TM}, E_{DFA}, E_{CFG}, E_{TM}, ALL_{DFA}, ALL_{CFG}, ALL_{TM}, EQ_{DFA}, EQ_{CFG}, EQ_{TM}.

ONLY IF you finish problems above, look at the busy beaver problem below. The problem is interesting, famous, and challenging. It is also *much, much* more complicated than any problem I would *ever* ask on a homework or exam. It's here for your intellectual enjoyment!

3. **Busy beaver.** (Sipser 5.16) This problem is a Theory of Computation classic. It explores the idea of "computable function" and asks: are there some *numerical* functions which are not computable?

   Let $\Gamma = \{a, b, \sqcup\}$ be the tape alphabet for all Turing machines in this problem.

   Define the set:

$$HALT_{\text{TM}}^{k,\varepsilon} = \{\langle M \rangle \mid \text{ deterministic Turing machine } M \text{ has } k \text{ states and halts on input } \varepsilon\}$$

   Let $BB : \mathbb{N} \to \mathbb{N}$ be a function defined as follows: $BB(k)$ is the maximum number of 'a's on the tape of any machine in $HALT_{\text{TM}}^{k,\varepsilon}$ after it halts on input ε. This is called the "busy beaver" function.

   Show that the function BB is not computable. This will be a proof by contradiction, in several steps:

   (a) Show that BB is a strictly increasing function: $BB(n) < BB(n+1)$.

   (b) Show that if $f : \mathbb{N} \to \mathbb{N}$ is a computable function, then there is some integer q such that $f(n) \leq BB(n+q)$.
   (Hint: design a machine with approximately q states that when started with input $w = a^n$ halts with output $a^{f(n)}$ on its tape.)

   (c) Use parts (a) and (b) to obtain a contradiction.
   (Hint: Assume for contradiction that $BB(n)$ is computable. Show that this implies that $g(n) = BB(2n)$ is computable. Use this to get a contradiction.)