CS 332: *Theory of Computation* Prof. Mark Bun
Boston University November 2, 2021

Homework 7 – Due Tuesday, November 2, 2021 at 11:59 PM

Reminder Collaboration is permitted, but you must write the solutions *by yourself without assistance*, and be ready to explain them orally to the course staff if asked. You must also identify your collaborators and write "Collaborators: none" if you worked by yourself. Getting solutions from outside sources such as the Web or students not enrolled in the class is strictly forbidden.

Note You may use various generalizations of the Turing machine model we have seen in class, such as TMs with two-way infinite tapes, stay-put, or multiple tapes. If you choose to use such a generalization, state clearly and precisely what model you are using. **You may describe Turing machines at a high-level on this assignment.**

Problems There are 4 required problems and 1 bonus problem.

1. (**Uncountable sets**)

   (a) Aliens from the planet Fubar'd have *(countably) infinite* single-strand DNA sequences from the set of nucleobases $\{A, C, G, T\}$. Let $\mathcal{D}$ be the set of all possible DNA sequences for residents of Fubar'd, so $\mathcal{D} = \{a_1 a_2 a_3 \cdots \mid a_i \in \{A, C, G, T\}, i \in \mathbb{N}\}$. Show that $\mathcal{D}$ is **uncountable**.

   (b) A function $f : \mathbb{N} \to \mathbb{N}$ is *increasing* if $f(i) < f(i+1)$ for every $i \in \mathbb{N}$. So $f(1) = 1, f(2) = 4, f(3) = 9, \ldots$ are the first few values of (what looks like) an increasing function f. But the function g where $g(1) = 2, g(2) = 1, g(3) = 7, \ldots$ is not increasing.

   Show that $\mathcal{F} = \{f : \mathbb{N} \to \mathbb{N} \mid f \text{ is increasing}\}$, the set of all increasing functions, is uncountable.

   Hint: When you construct a function contradicting the diagonal, make sure that it is indeed a member of $\mathcal{F}$, i.e., increasing.

2. (**Reduction Mad-Libs**) A language $B \subseteq \{\mathsf{y}, \mathsf{z}\}^*$ is *sleepy* if every string in B contains "zzz" as a substring. For example, the empty language, $\{\mathsf{z}^n \mid n \geq 3\}$, and $\{\mathsf{y}^n \mathsf{z}^m \mathsf{y}^n \mid m \geq 3, n \geq 0\}$ are all sleepy, but $\{\mathsf{y}, \mathsf{yz}, \mathsf{yzz}, \mathsf{yzzz}\}$ and $\{\mathsf{z}^n \mid n \geq 0\}$ are not sleepy. The language $S_{\text{TM}} = \{\langle M \rangle \mid L(M) \text{ is sleepy}\}$ corresponds to the following computational problem: Given the encoding of a TM M, does M recognize a sleepy language? This exercise will walk you through a proof, by reduction, that S_{TM} is undecidable.

   Assume, for the sake of contradiction, that S_{TM} is decidable by a TM R. That is, there is a TM R that accepts $\langle M \rangle$ whenever $L(M)$ is sleepy, and rejects $\langle M \rangle$ whenever $L(M)$ is not sleepy. We will use R to construct a new TM T that decides the (undecidable) language A_{TM}.

```
Algorithm: $T(\langle M, w \rangle)$
  Input : Encoding of a basic TM $M$ over input alphabet $\{y, z\}$, string $w \in \{y, z\}^*$
    1. Construct the following TM $N$:
         $N$ = "On input a string $x \in \{y, z\}^*$:
           If $x \neq$ zz, reject.
           Else, run $M$ on input $w$. If it accepts, accept. Otherwise, reject."
    2. Run $R$ on input $\langle N \rangle$. If it accepts, ___(i)___. Otherwise, ___(ii)___.
```

(a) This proof is by reduction from a language A to a language B. What are the languages A and B? (Make it clear in your solution which one is A and which one is B, since the order matters a lot!)

(b) Consider the machine N constructed inside algorithm T. If M accepts on input w, what is the language $L(N)$? Is $L(N)$ sleepy in this case?

(c) If M does not accept on input w, what is the language $L(N)$? Is $L(N)$ sleepy in this case?

(d) Fill in the blanks labeled (i) and (ii) with *accept* or *reject* decisions to guarantee the following conditions: If M accepts input w, then T accepts input $\langle M, w\rangle$, and if M does not accept input w, then T rejects input $\langle M, w\rangle$. Use parts (b) and (c) to explain why these conditions hold for your choices of how to fill in the blanks.

(Your job is done now, but you may want to keep reading to see the exciting conclusion of the proof.) By part (d), the TM M exactly decides the language A_{TM}. But this language is undecidable, which is a contradiction. Hence our assumption that S_{TM} was decidable is false, so we conclude that S_{TM} is an undecidable language.

3. (**Spooky TMs**) A two-tape Turing machine M *is spooky* if there exists a string $w \in \{0,1\}^*$ such that, on input w, the TM M has "1000" appear somewhere on its second tape when run on input w.[1] Consider the problem of determining whether (the encoding of) a TM M is spooky.

   (a) Formulate this problem as a language $\text{SPOOKY}_{\text{TM}}$. Caution: The only input to this computational problem is $\langle M\rangle$ for a TM M. This problem is closely related to, but for this reason different from, the language recognized by H in Discussion 7, Problem 2. Please make sure you understand how it's different.

   (b) Prove that the language $\text{SPOOKY}_{\text{TM}}$ is undecidable.

   Hint: Give a reduction from the undecidable language A_{TM}. That is, you should assume for the sake of contradiction that $\text{SPOOKY}_{\text{TM}}$ is decidable. Then under this assumption, construct a TM deciding A_{TM}, explain why your decider is correct, and as a result conclude that your assumption that $\text{SPOOKY}_{\text{TM}}$ is decidable must have been false. It's also fine if you want to give a reduction from a different undecidable language, instead, but your proof should still have this structure.

   (c) Is $\text{SPOOKY}_{\text{TM}}$ Turing-recognizable? Is $\overline{\text{SPOOKY}_{\text{TM}}}$ Turing-recognizable? Give a convincing explanation for both of your answers, but a complete description of a TM or of a reduction is not necessary.

4. (**Subset detection**) Consider the following computational problem: Given the (encodings of) two basic TMs M and N, determine whether the language recognized by M is a subset of the language recognized by N.

   (a) Formulate this problem as a language $\text{SUBSET}_{\text{TM}}$.

   (b) Prove that the language $\text{SUBSET}_{\text{TM}}$ is undecidable.

5. (**Bonus problem**) Define the language $XOR_{\text{TM}} = \{\langle M, w, v\rangle \mid M \text{ is a TM that accepts exactly one of the strings } w, v\}$. Prove that both XOR_{TM} and its complement $\overline{XOR_{\text{TM}}}$ are both unrecognizable.

[1] The string "1000" is supposed to look kind of like "boo." Happy Halloween!