CSE P531: Computability and Complexity Theory (Spring, 2016)

Homework 2 Out: Thursday, 8-Apr. **Due**: Saturday, 16-Apr (9pm in the **Dropbox**)

Reading:

Sipser, Sections 4.1-4.2 and 5.1-5.3

Instructions:

Your proofs and explanations should be clear, well-organized and as concise as possible.

You are allowed to discuss the problems with fellow students taking the class. However, you must write up your solutions completely on your own. Moreover, if you do discuss the problems with someone else, I am asking, on your honor, that you do not take any written material away from the discussion. In addition, for each problem on the homework, I ask that you acknowledge the people you discussed that problem with, if any.

Most of the problems require only one or two key ideas for their solution – spelling out these ideas should give you most of the credit for the problem even if you err in some finer details. So, make sure you clearly write down the main idea(s) behind your solution.

A final piece of advice: Begin work on the problem set early and don't wait until the deadline is only a few days away.

1. Useless states

A **useless state** in a Turing machine is one that is never entered on any input string. Consider the problem of determining whether a Turing machine has any useless states. Formulate this problem as a language and show that it is undecidable.

2. Mapping reductions

Recall the definition of a computable function (Section 5.3 of the book). A language A is **mapping-reducible** to a language B, written $A \leq_m B$ if there is a computable function $f : \Sigma^* \to \Sigma^*$ where for every $w \in \Sigma^*$,

$$w \in A \Leftrightarrow f(w) \in B$$

The function f is called a **reduction** of A to B. One can easily check that if $A \leq_m B$ and B is decidable, then A is decidable as well.

a) Consider the language

$L = \{ \langle M, w \rangle$: M is a TM that on input w ever attempts to move its tape head left when it is on the left-most cell of the tape $\}$.

Show that L is undecidable by giving a reduction from A_{TM} to L.

b) Give an example of an undecidable language L where $L \leq_m \overline{L}$.

3. Uncomputable numbers

Consider Turing machines with alphabet $\Sigma = \{0,1\}$ and tape alphabet $\Gamma = \{0,1,\blacksquare\}$ (here $\blacksquare$ is meant to be the blank symbol). Define the function $B : \mathbb{N} \to \mathbb{N}$ as follows. For each value k, consider all the k-state TMs that halt when started with a blank input tape. Let $B(k)$ be the maximum number of 1s that remain on the tape among all these machines.

a) Show that B is **not** a computable function.

b) Show that B grows faster than any computable function in the following sense: If $f : \mathbb{N} \to \mathbb{N}$ is computable, then for some value k we must have $B(k) \geq f(k)$. [This means that B grows super fast! Note that functions like $f(k) = 2^{2^{2^{2^{2^{\cdots}}}}}$ where there are k 2's in this tower **are** computable. $B(k)$ grows faster than any such function.]

Extra credit problems (both problems are worth TWO regular problems)

1. Let A and B be two disjoint languages. Say that a language C **splits** A and B if $A \subseteq C$ and $B \subseteq \bar{C}$. The language C here acts as a **proof** that A and B are disjoint. Describe two Turing-recognizable languages that cannot be split by any decidable language.

2. Suppose that we change the Turing machine model so that it cannot alter the input (the input symbols must remain in place), but it can write anywhere else on the tape. Give a complete proof that such a Turing machine cannot even recognize the language of palindromes over the alphabet $\{0,1\}$.