# CSC 341 - Fall 2022
# Problem Set 7 Solutions

**Problem 1.** We defined the language $A_{TM}$ to be the set of $\langle M, \omega \rangle$ such that $M(\omega)$ accepts. Prove that $\overline{A_{TM}}$ is not recognizable. You may assume that $A_{TM}$ is not *decidable*.

**Solution 1.** We know that $A_{TM} \in RE$ and that $A_{TM} \notin DEC$. Thus $A_{TM} \notin co{-}RE$, and therefore $\overline{A_{TM}} \notin RE$. $\qquad\square$

**Problem 2.** The set of all strings, $\Sigma^*$, is countable. This means there is a (computable) bijection $\phi : \mathbb{N} \to \Sigma^*$ that enumerates $\Sigma^*$. As a consequence, this readily yields a notion of subsets of the natural numbers being recognizable.

Let $L \subseteq \mathbb{N}$. We say that $L$ is *recognizable* if there is a Turing machine $M$ for which

$$L(M) = \{\phi(n) \mid n \in L\}.$$

Generally, we leave $\phi$ implicit and simply allow $M$ to accept a natural number input $n$.

Give a definition of recognizability for any countable set $S$ and prove that it is equivalent to the definition for $\mathbb{N}$.

**Solution 2.** This could be considered a double trick question. Let $S$ be a countable set. Then there is some bijection $\psi : S \to \Sigma^*$. Consider the case $S = A_{TM}$. If $\phi$ were computable, then the language $\Sigma^*$ is decidable, which would make $\psi^{-1}(\Sigma^*) = A_{TM}$ decidable as well. However, there is a key point to recognize here: we don't ever compute the symbols in $S$. We are not required to compute $\psi$ at any point. The elements of $S$ are just labels, and if the labels happen to look like something not computable, try not to look at them so hard.

We now give a definition of recognizability. Let $S$ be a countable set. If $S$ is finite, then it is decidable, so assume $S$ is infinite. Then there is some bijection $\psi : S \to \Sigma^*$. We say that a subset $S' \subseteq S$ is *recognizable* if there is a Turing machine $M$ for which

$$L(M) = \{\psi(s) \mid s \in S'\}.$$

To see why this definition is equivalent to the definition of recognizability, first notice that $\mathbb{N}$ is a countable set. As such, if $N \subseteq \mathbb{N}$ is recognizable with respect to the natural numbers definition, then it is with respect to the general definition as well.

Next, suppose $S' \subseteq S$ is recognizable. Now consider the set $A = \psi(S')$. We know that there is a TM $M$ which recognizes the language $A$. But then $M$ also recognizes the language $\phi(\phi^{-1}(A)) = A$, so $\phi^{-1}(A) \subseteq \mathbb{N}$ is recognizable. $\qquad\square$

**Problem 3.** Prove that the language

$$S = \{(i, j) \mid M_i(j) \text{ accepts}\}$$

is recognizable but not decidable, where $M_i$ is the $i^{th}$ Turing machine and $j$ corresponds to the $j^{th}$ string in $\Sigma^*$ (see problem 2).

**Solution 3.** Pick $\psi$ to be the bijection which maps $\mathbb{N} \times \mathbb{N}$ to $\Sigma^*$ in the definition of recognizability. Since we know $A_{TM}$ is recognizable but not decidable, it suffices to show that there is a (in this case) *computable* bijection $\phi$ between $\psi(S)$ and $A_{TM}$ such that $\omega \in \psi(S)$ iff $\phi(\omega) \in A_{TM}$.

There is some computable (and reversible) bijective encoding $E$ that maps pairs of Turing machine descriptions with an input string to strings in $\Sigma^*$ for $\psi(S)$. There is another one for $A_{TM}$. Call this one $E'$. Then we can map $\psi(S)$ to $A_{TM}$ via $E'(E^{-1}(\psi(S))) = A_{TM}$. $\qquad\square$

**Problem 4.** Prove that the language

$$K = \{i \mid M_i(i) \text{ halts}\}$$

is recognizable (see problem 2).

Now prove that the language

$$\overline{K} = \{i \mid M_i(i) \text{ does not halt}\}$$

is not recognizable (i.e. $K$ is not co-recognizable).

**Solution 4.** We first prove that $K$ is recognizable. To do so, we give a Turing machine $M$ for it.

$M =$ "On input $i$,

1. Run $M_i$ on the $i^{th}$ string of $\Sigma^*$
2. Accept"

Clearly, $M$ accepts $i$ if and only if the $i^{th}$ Turing machine halts on the $i^{th}$ string of $\Sigma^*$, so $L(M) = K$.

We now show that $\overline{K}$ is not recognizable. To do so, we give a mapping reduction $f$ from $S$ (from the previous problem which is not co-recognizable) to $K$ so that $(i, j) \in S$ iff $f(i, j) \in K$.

Define the TM $N_{i,j}$ to be the TM which on input $k$ runs $M_i(j)$ and accepts if $M_i(j)$ accepts and loops forever otherwise. In other words, $N_{i,j}$ halts iff $M_i(j)$ accepts. Since $N_{i,j}$ is a TM, there is some computable function $\sigma$ such that for all $(i, j)$, $N_{i,j}$ is the $\sigma(i,j)^{th}$ TM. Then if $f(i, j) = \sigma(i, j)$, we have that $(i, j) \in S$ iff $\sigma(i, j) \in K$ as desired. Thus $S \leqslant_m K$, and since $S$ is not co-recognizable, it follows that $K$ is not co-recognizable, hence $\overline{K}$ is not recognizable. $\qquad\square$

**Problem 5.** Prove that the language

$$FIN_{TM} = \{i \mid |L(M_i)| < \infty\}$$

is neither recognizable nor co-recognizable (see problem 2). (Hint: Both results can arise from a reduction from $\overline{K}$. One is easy. The other is not.)

**Solution 5.** We'll prove both $\overline{K} \leqslant_m FIN_{TM}$ and $\overline{K} \leqslant_m \overline{FIN_{TM}}$. Since $\overline{K}$ is not recognizable, it will then follow that $FIN_{TM}$ is neither recognizable nor co-recognizable. Before proceeding, let $k$ be the computable function that maps Turing machines to $\mathbb{N}$.

We first prove $\overline{K} \leqslant_m FIN_{TM}$. Define the TM $N_i$ to be as follows.

$N_i =$ "On input $j$,

1. Run $M_i(i)$

2. Accept"

Notice that $L(N_i) = \varnothing$ if $M_i(i)$ doesn't halt and $L(N_i) = \Sigma^*$ if $M_i(i)$ does halt. If we define the function $f(i) = k(N_i)$, then we have the following. $i \in \overline{K}$ iff $M_i(i)$ does not halt. But $L(N_i)$ is finite iff $M_i(i)$ does not halt. So we have $i \in \overline{K}$ iff $f(i) \in FIN_{TM}$.

We now prove that $\overline{K} \leqslant_m \overline{FIN_{TM}}$. Define the TM $N_i$ to be as follows.

$N_i =$ "On input $j$,

1. Run $M_i(i)$ for $j$ steps

2. Accept if $M_i(i)$ did not halt and reject otherwise"

If $M_i(i)$ halts, then it does so in some finite number of steps $j^*$. In this case, $|L(N_i)| = j^* - 1$. However, if $M_i(i)$ does not halt, then $N_i$ accepts every input. So we have that $L(N_i)$ is infinite iff $M_i(i)$ does not halt. Thus if we define the function $f(i) = k(N_i)$, we get that $i \in \overline{K}$ iff $M_i(i)$ does not halt. This gives us that $i \in \overline{K}$ iff $f(i) \in \overline{FIN_{TM}}$. $\qquad\square$