CS 334 ~ Problem Set #8:

[illegible] "I pledge my honor I have abided by the Stevens Honor System".
Partner: Hiya Bhavsar

#1) Show DECIDERS = $\{\langle M \rangle : M$ encodes a TM that halts on every input$\}$ is not turing-recognizable.

Assume (for contradiction) DECIDERS (abreviated D) is TM-Recognizable. Then there must exist an enumerator for D, call it $\mathcal{E}_D$. Create a filter for $\mathcal{E}_D$ which rejects repeated prints, & pipe it into $\mathcal{E}_D'$, so $\mathcal{E}_D'$ prints unique $\langle M \rangle$ for D.

Suppose we have a set of every possible input string for the accepted alphabet of D, Σ. For clarity we can use $\Sigma = \{0, 1\}$, $\rightarrow \alpha = \{\varepsilon, 0, 1, 00, 01, 10, 11, \ldots\}$ but realize we can generalize this set to any alphabet (since their size is finite).

Notice we can make a bijection from $\alpha \rightarrow \mathbb{N}$, where
$\alpha_1 = \varepsilon$, $\alpha_2 = 0$, $\alpha_3 = 1$, $\alpha_4 = 00$, ...
Thus the string's index maps to the naturals.

Now we use diagnolization to show such mapping is impossible for D. Construct a TM S.

S: On input α_i,
1. Run α_i on the i^{th} output of $\mathcal{E}_D'$.
2. If accepts, reject. If rejects, accept.

Since every $\langle M \rangle$ in D is a decider, α_i will always either reject or accept. However, we've created a contradiction. Since α spans every possible input on $\langle M \rangle$ you can imagine the results of running α on any $\langle M_i \rangle$ as the following.

Running α on $\langle M_i \rangle$

	α_1	α_2	α_3	...
$\langle M_1 \rangle$	A	A	R	
$\langle M_2 \rangle$	R	R	A	...
$\langle M_3 \rangle$	A	R	A	
...	...	...		...

where A is accept
R is reject

This table should contain the results of every possible decider M. But we just created a decider S that accepts or rejects the opposite of exactly 1 input from every $\langle M_i \rangle$. So S is not in D, which is a contradiction.

$\Downarrow$

	α_1	α_2	α_3	...
$\langle M_1 \rangle$	R	A	R	
$\langle M_2 \rangle$	R	A	A	...
$\langle M_3 \rangle$	A	R	R	
...	...	...		...

S

$\boxed{\text{Thus D is unrecognizable.}}$ ∎

#2) Assume $Left_{TM}$ is a decidable language for sake of contradiction. Then there exists a decider for $Left_{TM}$, call it H. We'll use H to construct A_{TM} decider from $Left_{TM}$, thus showing it is undecidable.

→ First we construct a helper TM, M_1, s.t

M_1: on input w

1. M_1 will simulate M on w, but M_1 will mark the left most cell first cell on the tape. If M ever reaches the leftmost cell & goes left again, we'll have M_1 go right then back left, so that it ends up in the same state as M, however it does not hit the leftmost symbol. (This is so H doesn't mistakenly accept M)
2. If M accepts, have M_1 move left until it hits its marked position. Then move left once more. If M rejects, reject w/o moving left from our marked state.

This way if $M(w)$ accepts, we move left from the leftmost. Otherwise we don't.

Now we construct N as follows.

N: on input M, w where
M is a TM description & w is its input string

1. Construct the TM M_1.
2. Run $H(M_1, w)$. If we accept, then accept.
   Else, we reject.

Since we constructed M_1 in such a way that if M accepts on input w, M will move left from the leftmost position, otherwise it won't. H becomes reduced to deciding whether our Tm M accepts some string w, which is exactly A_{TM}.

Thus $A_{TM} \leq Left_{TM}$, implying $Left_{TM}$ is undecidable. ▨

#3) Let $L = \{ \langle M, w, k \rangle :$ TM M accepts input w and never moves its head beyond the first k tape cells $\}$

Show L is decidable.

We start by showing L is recognizable.

Construct a TM Z to recognize L.

Z: input M, w, k (M is a TM, w is a string, k is an int)
1. Run M(w) by 1 step.
2. If M(w) passes position k, (ends in k+1 or larger) reject.
   If M(w) rejects, reject.
   If M(w) accepts, accepts.
3. Repeat from step 1.

Since we've constructed a TM Z for L, L is turing Recognizable.

Next we show $\overline{L}$ is turing recognizable.

$\overline{L} = \{ \langle M, w \rangle \mid M$ rejects $w \lor M$'s head passes the k^{th} tape cell.

We construct a TM E, that recognizes $\overline{L}$.

E: On input M, w, k (M is a TM, w is a string, k is an int)

1. Run $M(w)$ by 1 step.
2. If $M(w)$ is passed position k, accept.
   If $M(w)$ is in reject state, accept.
   If $M(w)$ is in accept, reject.
3. Repeat from step 1.

Since we're constructed a TM E for $\overline{L}$, $\overline{L}$ is turing recognizable.

Now we've shown both L & $\overline{L}$ are turing recognizable. For any language A if A & its compliment are turing recognizable, A is turing decidable. Thus L is turing decideable.

$\blacksquare$