CIS 2620 48621473

Homework 1

Collaborators: Ethan Yu

Q1. To prove that L is not recursively enumerable, we can indeed reduce it from the language $\overline{\text{A}_{TM}}$. Using the input $\langle M \rangle, w$ for $\overline{\text{A}_{TM}}$, we define the input for L as $\langle M' \rangle, x, y$ such that $x = w$, $y = s$, where $s \neq w$ and s is an input for $\langle M \rangle$. Let us now define $\langle M' \rangle$.

Define $M'(k)$ as follows:

(a) If k is not equal to w, accept k.

(b) Otherwise, run the Universal Turing Machine (UTM) on $\langle M \rangle$ and k.

(c) If the Universal Turing Machine accepts, accept k.

(d) Otherwise, reject k.

Let us prove the validity of this reduction:

$\Rightarrow$ If $\langle M \rangle, w \in \overline{\text{A}_{TM}}$, then $\langle M' \rangle, x, y \in L$.

First of M' will always accept y as by line 1 $y \neq w$. Let us show that it will not accept x. By definition, since $\langle M \rangle, w \in \overline{\text{A}_{TM}}$, M will not accept w and also x as $x = w$. When we feed x in M', we will go to line 2 as $x = w$. From there, by defitnton of UTM, the UTM will reject M and x as M does not accept x. Therefore, we will go to line 4, and reject x. Hence M' will always reject x but accept y, meaning only one of x or y is accepted, meaning $\langle M' \rangle, x, y \in L$.

$\Leftarrow$ If $\langle M' \rangle, x, y \in L$, then $\langle M \rangle, w \in \overline{\text{A}_{TM}}$

This means that M' accepted only x or only y. $y = s$ and $x = w$. Only of those will be accepted. Since $y \neq w$, we see that y will always be accepted by M'. This means $x = w$ will always be rejected. Our only rejection is line 4. We reach line 4 through line two where we run the UTM on $\langle M \rangle$ and our input string, in this case x. For our to reach line 4, our UTM much have rejected $\langle M \rangle, w = x$. By definition, this means thats M does not accept $w = x$. By definition of $\overline{\text{A}_{TM}}$, this means that $\langle M \rangle, w \in \overline{\text{A}_{TM}}$

We have proved that $\langle M' \rangle, x, y \in L$, iff $\langle M \rangle, w \in \overline{\text{A}_{TM}}$ through our vald reduction. As $\overline{\text{A}_{TM}}$ is not recursively enumerable, this means that L is not either.

Q2:

Prove that $L \leq A_{TM}$ if and only if L is recursively enumer- able.

$\rightarrow$ If $L \leq A_{TM}$, then L is recursively enumerable.
From lecture notes, we know that if $L_i \leq L_j$ and L_j is recursively enumerable, then L_i is also recursively enumerable. From lecture notes, we know that A_{TM} is recursively enumerable; since $L \leq A_{TM}$, this means that L is recursively enumerable using the lecture statement we showed earlier.

$\leftarrow$ If L is recursively enumerable, then $L \leq A_{TM}$, . We know that L is recursively enumerable so there must exists a Turing machine M such that L(M) = L. Let us define the following reduction from L to A_{TM}. For an input w of the Turing Machine M for L, let $N = M$ and $y = w$, where $\langle N \rangle$, y is the input for A_{TM}. Let us show that this is a valid reduction below and $L \leq A_{TM}$.

SUB PROOF: $w \in L($ M accepts $w)$ iff $\langle N \rangle$, $y \in A_{TM}$

$\rightarrow$ If $w \in L$ (M accepts w), then $\langle N \rangle$, $y \in A_{TM}$ $N = M$ and $y = w$.
Since, M accepts w this also means that N accepts y. Hence, $\langle N \rangle$, $y \in A_{TM}$.

$\leftarrow$ If $\langle N \rangle$, $y \in A_{TM}$, then $w \in L(M$ accepts $w)$. Since $\langle N \rangle$, $y \in A_{TM}$, this means that N accepts y, Since $N = M$ and $y = w$, this also means M accepts w and so $w \in L$.

Therefore our reduction is valid, meaning $L \leq A_{TM}$.

Q3: To prove that L_p is not recursively enumerable, we can reduce it from the language $\overline{\mathrm{A}_{TM}}$. We define the input for L_p as $\langle M' \rangle$, using the input $\langle M \rangle, w$ for $\overline{\mathrm{A}_{TM}}$. Now, let us define $M'(k)$ as follows:

Define $M'(k)$ as follows: For input string k

i. Run the universal turing machine U_{TM} on $\langle M \rangle$, w.

ii. If U_{TM} accepts, then reject k

iii. If not, run U_{TM} on $\langle M_0 \rangle, s$

iv. If U_{TM} accepts $\langle M_0 \rangle, s$, accept k

v. If not, reject k

Let us prove the validity of this reduction:

$\rightarrow$ If $\langle M \rangle, w \in \overline{\mathrm{A}_{TM}}$, then $\langle M' \rangle \in L_p$.
Since $\langle M \rangle, w \in \overline{\mathrm{A}_{TM}}$, M rejects w. This means that U_{TM} rejects $\langle M \rangle$, w. This means that we go to lines 3-5. Note that for every x, that M_0 accepts, M' will also accept as UTM will accept $\langle M_0 \rangle, x$ by definition. Likewise, for every z, that M_0 doesn't accept, M' will not accept either as UTM will not accept $\langle M_0 \rangle, z$ by definition. As such, the langauge of $L(M_0)$ and $L(M')$ are identical and since $L(M_0) \in P$, then $L(M') \in P$, meaning that $\langle M' \rangle \in P$

$\rightarrow$ If $\langle M' \rangle \in L_p$, then $\langle M \rangle, w \in \overline{\mathrm{A}_{TM}}$
Suppose for the sake of contradiction that $\langle M \rangle, w \notin \overline{\mathrm{A}_{TM}}$. As such, M accepts w, meaning UTM will also accept $\langle M \rangle, w$, meaning our M' will always be in line 2 and reject every string. Thus, $L(M')$ is the empty set, which is a finite subset of $L(M_0)$. As such, $L(M')$ does not have the property P, meaning, $\langle M' \rangle \notin L_p$, which is a contradiction.

We have proved that $\langle M \rangle, w \in \overline{\mathrm{A}_{TM}}$ iff $\langle M' \rangle \in L_p$ through our vald reduction. As $\overline{\mathrm{A}_{TM}}$ is not recursively enumerable, this means that L_p is not either.