CS 4820, Spring 2018	**Homework 10, Problem 3**
Name: Rongguang Wang	
NetID: rw564	
Collaborators: Siyuao Liu (sl2928); Yihao Chen (yc2288)	

(3) *(10 points)* As in question (2), let $T \subset \Sigma^*$ denote the set of all strings x such that x is the description of a termination-safe Turing machine. Prove that $\overline{T}$, the complement of T in Σ^*, is not recursively enumerable.

Solution:

The problem is aiming to prove that the set $\bar{T}$ is not recursively enumerable ($r.e.$), where T belongs to Σ^* denote the set of all strings x such that x is the description of a termination-safe Turing Machine. Termination-safe Turing Machine refers to one that is guaranteed to terminate on every input string. To prove that set $\bar{T}$ is not recursively enumerable, we reducing from the co-halting problem $\bar{H}$ to $\bar{T}$ since $\bar{H}$ is not recursively enumerable. Given a Turing Machine $M_{\bar{T}}$ which is used to solve the termination-safe problem such that $M_{\bar{T}}(M)$ outputs YES if and only if Turing Machine M does not halt on input y. We need another machine C which is used to solve the co-halting problem such that $C(x; y)$ outputs YES if and only if x does not halt on input y. Then, we can construct machine C which encoding machine x and input y as the input of machine $M_{\bar{T}}$. Assume that machine $M_{\bar{T}}$ takes in a machine which is the encapsulating of x and y which represents as $S_{x,y}$. There is a one-to-one relationship which combines machine $M_{\bar{T}}$ and machine C that Turing Machine M is not termination-safe if and only if x does not halt on input y. Thus, Turing Machine $S_{x,y}$ should operates as following: If x is not halt on input y, then the machine should loop forever which means that $S_{x,y}$ is not termination-safe. If x halts on input y, then the machine halts which means that $S_{x,y}$ is termination-safe. Now we define the operation rule of Turing Machine $M_{\bar{T}}(S_{x,y})$. $M_{\bar{T}}(S_{x,y})$ outputs NO if and only if $S_{x,y}$ is not termination-safe which means that machine x is not halt on input y. $M_{\bar{T}}(S_{x,y})$ outputs YES if and only if $S_{x,y}$ is termination-safe which means that machine x is halt on input y. Therefore, we reduced $\bar{T}$ from the co-halting problem $\bar{H}$ which means that the set size of $\bar{T}$ is greater than or equals to the set size of $\bar{H}$. Since co-halting problem $\bar{H}$ is not recursively enumerable, $\bar{T}$ is not recursively enumerable. Next, we proof by strong induction. $x; y$ belongs to set $\bar{H}$ if and only if $M_{\bar{T}}$ accepts $x; y$ which means $S_{x,y}$ accepts $x; y$. Thus, $\bar{H}$ is recursively enumerable which contradict to the fact that $\bar{H}$ is not recursively enumerable. Therefore, $\bar{T}$ is not recursively enumerable.