Recursion Theorem

a Describe the following functions q and c at a high level in your own words:

Let $q(w)$ be computed by
$Q =$ "On input w:

- Construct TM $P_w =$ "On any input:
  - Erase the input.
  - Write w on the tape.
  - Halt"
- *Output* $\langle P_w \rangle$"

Let $c(X, Y)$ be computed by
$C =$ "On input $\langle X, Y \rangle$ where X, Y are TMs:

- Construct TM $M_{X,Y} =$ "On input w:
  - Run X on w to get a
  - Run Y on s to get b
  - *Output* b"
- *Output* $\langle M_{X,Y} \rangle$"

b Let B be the TM "On input $\langle M \rangle$, where M is a TM:

- Run Q on $\langle M \rangle$ to get $\langle R \rangle$
- Run C on $\langle R, M \rangle$ to get $\langle T \rangle$
- *Output* $\langle T \rangle$"

Let TM $G = c(q(B), B)$. Before worrying about what B and G *do*, first check that all the functions and TMs above seem well-defined. In particular, confirm that Q and C always halt (so that q and c are truly functions over the whole intended domains), and that there are no circular dependencies (the goal here is to demonstrate that recursion is legit, so we'd better not be *already* using recursion: no TM should be defined in terms of itself, nor should there be a larger cycle like E saying "Run F" and F saying "Run E"). Finally, trace through the computation of G on input 01; what is the output?

c A modification of the construction from the previous parts allows us to build TMs that can write out and then work with their own descriptions. The next few problems apply this new power.

Fix an encoding scheme $\langle\cdot\rangle$. We will say TM M is *minimal* if there is no TM R where $L(R) = L(M)$ and $|\langle R\rangle| < |\langle M\rangle|$. Let $MIN_{TM} = \{\langle M\rangle \mid M \text{ is a minimal TM}\}$. Complete the following proof that every infinite subset of MIN_{TM} is unrecognizable:

> Let L be an infinite subset of MIN_{TM}, and assume towards contradiction that L is recognizable. Then let E enumerate L. Now define TM C = "On input w:
>
> - Obtain self-description $\langle C\rangle$ (using the construction from the Recursion Theorem)
> - $\cdots$ (do stuff involving w, $\langle C\rangle$, and E)
>
> $\cdots$ then since it was printed by E it must be minimal, yet C has the same language and a shorter encoding, contradiction.

e For the following theorem, assume that we are using an encoding scheme $\langle \cdot \rangle$ for which *every* string is a valid TM encoding. (For example, start with any other encoding scheme, and then declare that all strings which are not valid encodings under the old scheme become valid encodings of "On input w: *Reject*" in the new scheme.)

(from Sipser Theorem 6.8): For any computable function $t : \Sigma^* \to \Sigma^*$, there exists a TM F where $L(F) = L(t(\langle F \rangle))$.

i Demonstrate the claim holds for the function $t_1(\langle X \rangle) = \langle$"On input w: Run X on $0w$ and *accept/reject* as it does$\rangle$, by finding an F where $L(F) = L(t_1(\langle F \rangle))$

ii A student claims that each of the following functions serves as a counterexample to the theorem. Explain why they're wrong.

- $t_2(\langle X \rangle) = \langle X$ with the accept and reject states switched$\rangle$
- $t_3(\langle X \rangle) = \begin{cases} \langle \text{"On input } w\text{: } \textit{Accept}\text{"} \rangle, & \text{if } L(X) = \emptyset \\ \langle \text{"On input } w\text{: } \textit{Reject}\text{"} \rangle, & \text{otherwise} \end{cases}$

iii Prove the theorem. *(Hint: Construct an F which makes use of $\langle F \rangle$.)*

d Use our Recursion Theorem powers to prove that E_{TM} is undecidable. *(Do not appeal to Rice's Theorem or to the undecidability of another language like A_{TM} - you can arrive at a contradiction very directly using a TM that has access to its own description.)*