8.4

Languages defined by a Turing machine
Recursive vs. Recursively Enumerable

1. **Recursively enumerable** (aka **RE**) languages

   \[ L = \{ L(M) \mid M \text{ some Turing machine} \} . \]

2. **Recursive / decidable** languages

   \[ L = \{ L(M) \mid M \text{ some Turing machine that halts on all inputs} \} . \]

3. Fundamental questions:
   1. What languages are RE?
   2. Which are recursive?
   3. What is the difference?
   4. What makes a language decidable?
   5. How much wood would a TM chuck, if a TM could chuck wood?
Recursive vs. Recursively Enumerable

1. **Recursively enumerable (aka RE) languages (bad)**
   \[ L = \{ L(M) \mid M \text{ some Turing machine} \} . \]

2. **Recursive / decidable languages (good)**
   \[ L = \{ L(M) \mid M \text{ some Turing machine that halts on all inputs} \} . \]

Fundamental questions:
1. What languages are RE?
2. Which are recursive?
3. What is the difference?
4. What makes a language decidable?
5. How much wood would a TM chuck, if a TM could chuck wood?
Recursive vs. Recursively Enumerable

1. **Recursively enumerable** (aka \textit{RE}) languages (bad)
   \[ L = \{ L(M) \mid M \text{ some Turing machine} \} . \]

2. **Recursive / decidable** languages (good)
   \[ L = \{ L(M) \mid M \text{ some Turing machine that halts on all inputs} \} . \]

3. Fundamental questions:
   1. What languages are RE?
   2. Which are recursive?
   3. What is the difference?
   4. What makes a language decidable?
   5. How much wood would a \textit{TM} chuck, if a \textit{TM} could chuck wood?

Har-Peled (UIUC) CS374 Fall 2020
How was the Turing Machine invented...

GREAT. A WAREHOUSE FILLED WITH MILES AND MILES OF REWRITABLE TAPE! WHAT ARE WE EVER GOING TO DO WITH THIS, ALAN?

...ALAN?

And thus the Turing Machine was born.
THE END

... 

(for now)