

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)**

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

- 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)**

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

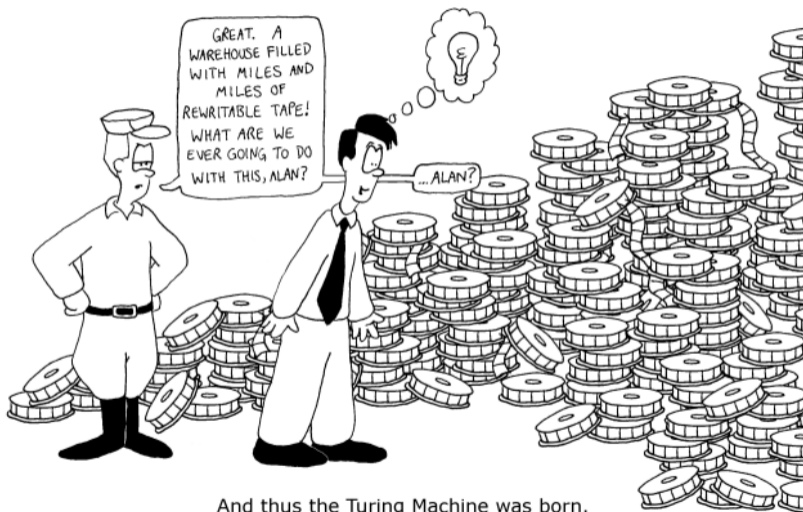
- 2 Recursive / decidable languages **(good)**

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

- 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?

How was the Turing Machine invented...



THE END

...

(for now)