## Week8 monday

is no	ot Turing-decidal	ole.
	is no	is not Turing-decidal

**Proof**: Suppose towards a contradiction that there is a Turing machine that decides  $A_{TM}$ . We call this presumed machine  $M_{ATM}$ .

By assumption, for every Turing machine M and every string w

- If  $w \in L(M)$ , then the computation of  $M_{ATM}$  on  $\langle M, w \rangle$  \_\_\_\_\_\_
- If  $w \notin L(M)$ , then the computation of  $M_{ATM}$  on  $\langle M, w \rangle$  \_\_\_\_\_\_

Define a **new** Turing machine using the high-level description:

D = "On input  $\langle M \rangle$ , where M is a Turing machine:

- 1. Run  $M_{ATM}$  on  $\langle M, \langle M \rangle \rangle$ .
- 2. If  $M_{ATM}$  accepts, reject; if  $M_{ATM}$  rejects, accept."

Is D a Turing machine?

Is D a decider?

What is the result of the computation of D on  $\langle D \rangle$ ?

<b>Theorem</b> (Sipser Theorem 4.22): A language is Turing-decidable if and only if both it and its complement are Turing-recognizable.
<b>Proof, first direction:</b> Suppose language $L$ is Turing-decidable. WTS that both it and its complement are Turing-recognizable.
<b>Proof, second direction:</b> Suppose language $L$ is Turing-recognizable, and so is its complement. WTS that $L$ is Turing-decidable.
Give an example of a <b>decidable</b> set:
Give an example of a <b>recognizable undecidable</b> set:
Give an example of an <b>unrecognizable</b> set:

