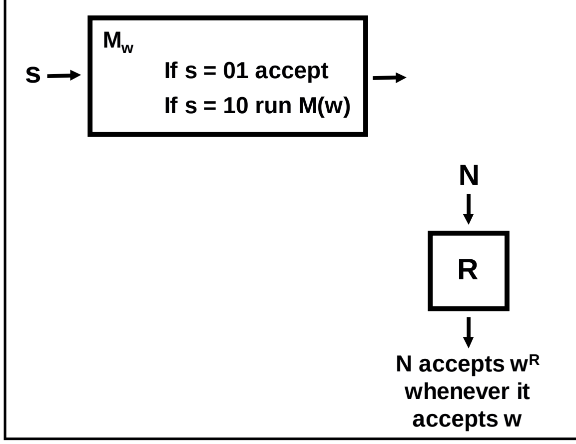
RICE'S THEOREM, THE RECURSION THEOREM AND THE FIXED-POINT THEOREM

THURSDAY OCT 13

Problem 1

Let S = { M | M is a TM that accepts w^R whenever it accepts w}. S is undecidable.



Problem 2.1 **UNDECIDABLE**

{ (M,w) | M is a TM that on input w attempts to move its head left from the leftmost cell }

Problem 2.2 **DECIDABLE**

{ (M,w) | M is a TM that on input w moves its head left at least once}

Problem 2.1 UNDECIDABLE

{ (M,w) | M is a TM that on input w attempts to move its head left from the leftmost cell }

Proof: Assume, for a contradiction, that TM L decides the language

We use L to decide A_{TM}

On input (M,w), make new machine N that marks the first square of the tape and then simulates M(w). If M moves to the marked square, N moves the head back to the second square. If M accepts, N moves its head all the way to the left

Run L on input N

Problem 2.2 DECIDABLE

{ (M,w) | M is a TM that on input w moves its head left at least once}

On input (M,w), run the machine for $|Q_M| + |w| + 1$ steps:

Accept	**If M moves its head to the left**
Reject	**Otherwise**

Problem 3

Let P be a language of Turing machines. Assume that P satisfies the following properties:

- **For any TMs M_1 and M_2, where $L(M_1) = L(M_2)$, $M_1 \in P$ if and only if $M_2 \in P$**
- **There exists TMs M_1 and M_2, where $M_1 \in P$ and $M_2 \notin P$**

Prove that P is undecidable

Theorem: P is undecidable

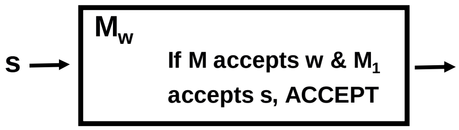
Proof: Assume, for a contradiction, that TM T_P decides P

Since P is decidable, $\neg$P is also decidable

Assume, WLOG, that P does not contain $T_\varnothing$ with $L(T_\varnothing) = \varnothing$

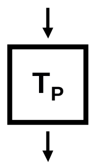
Let $M_1 \in P$

Algorithm for A_{TM}: On input (M,w):



If M accepts w, then $L(M_w) = L(M_1)$ and T_P accepts M_w

If M does not accept w, then $L(M_w) = \emptyset$ and T_P rejects M_w



RICE'S THEOREM

Let P be a language of Turing machines. Assume that P satisfies the following properties:

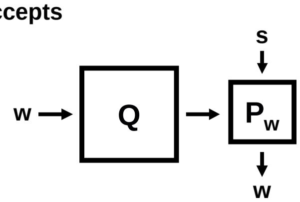
- **For any TMs M_1 and M_2, where $L(M_1) = L(M_2)$, $M_1 \in P$ if and only if $M_2 \in P$**
- **There exists TMs M_1 and M_2, where $M_1 \in P$ and $M_2 \notin P$**

Then P is undecidable

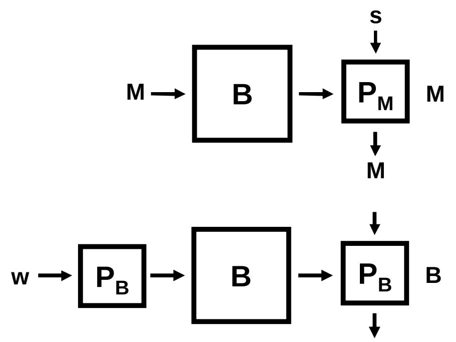
EXTREMELY POWERFUL!

The rest of the content of today's lecture has been a major source of headaches and misunderstandings

Theorem: There is a computable $q : \Sigma^* \to \Sigma^*$, where for any string w, q(w) is the description of a TM P_w that on any input, prints out w and then accepts



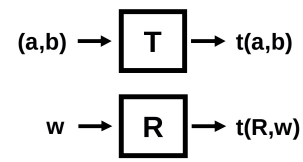
A TM THAT **PRINTS ITSELF**



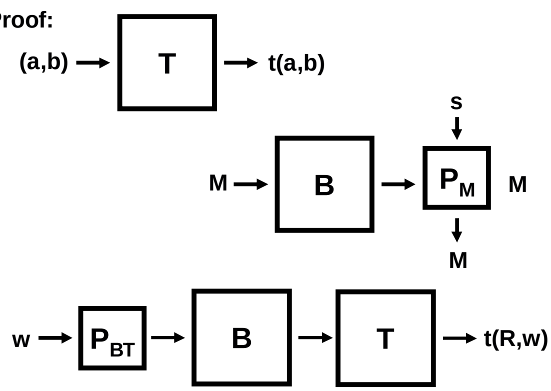
THE RECURSION THEOREM

Theorem: Let T be a Turing machine that computes a function $t : \Sigma^* \times \Sigma^* \rightarrow \Sigma^*$. There is a Turing machine R that computes a function $r : \Sigma^* \rightarrow \Sigma^*$, where for every w

$$r(w) = t(R, w)$$



Proof:



A Turing machine can obtain its own description and then go on to compute with it

Theorem: A_{TM} is undecidable

Proof (using the recursion theorem):

Assume H decides A_{TM}

Construct machine B such that On input w:

1. Obtains, via the recursion theorem, its own description B
2. Runs H on (B,w) and flips the output

Running B on input w does the opposite of what H says it should!

THE FIXED-POINT THEOREM

Theorem: Let t : $\Sigma^* \rightarrow \Sigma^*$ be a computable function. There is a Turing machine F such that t(F) describes a Turing machine equivalent to F.

Proof:

On input w:

1. Obtain, via the recursion theorem, its own description F
2. Compute t(F) to obtain a description of a TM G
3. Run G on w

WWW.FLAC.WS

Read chapter 6.1 of the book for next time