UNDECIDABILITY

Undecidable — NO Turing Machine decides the language

<u>Halting Problem</u>:

Given code $\langle M \rangle$ and x, does M halt on x?

Last time, we proved: HALTING PROBLEM IS UNDECIDABLE.

$$L_{SELFNOTHALT} = \{ \langle M \rangle \mid M \text{ on input } \langle M \rangle \text{ does not halt} \}$$

If $L_{SELFNOTHALT}$ were decidable, P_{SNH} would exist.

P_{SNH}

$\langle M \rangle \rightarrow$

output YES
when M on input $\langle M \rangle$
doesn't halt

output NO
otherwise

$M = P_{SNH}$

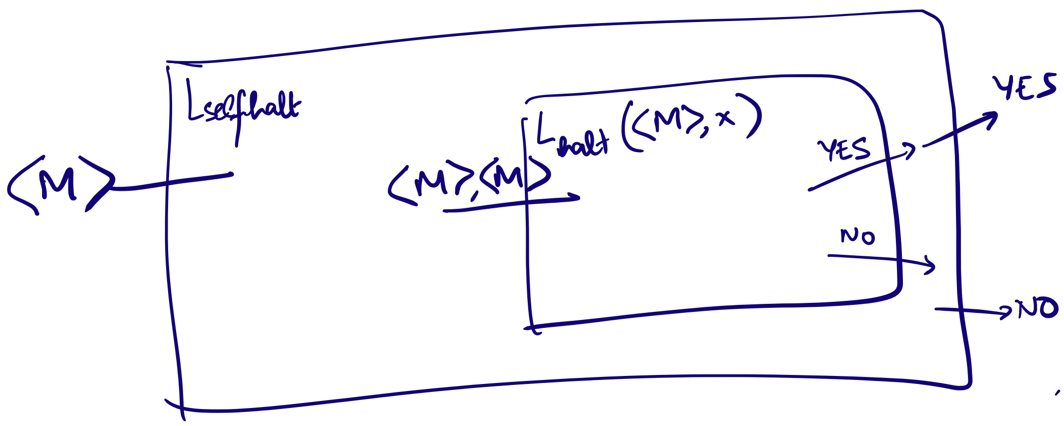
$P_{SNH}\langle M \rangle$ says YES (and halts!) $\iff$ M on input $\langle M \rangle$ doesn't halt.

$M = P_{SNH}$

$P_{SNH}\langle P_{SNH} \rangle$ says YES and halts $\iff$ P_{SNH} on input $\langle P_{SNH} \rangle$ doesn't halt

$L_{SELFHALT} = \{ \langle M \rangle \mid M$ on input $\langle M \rangle$ halts $\}$

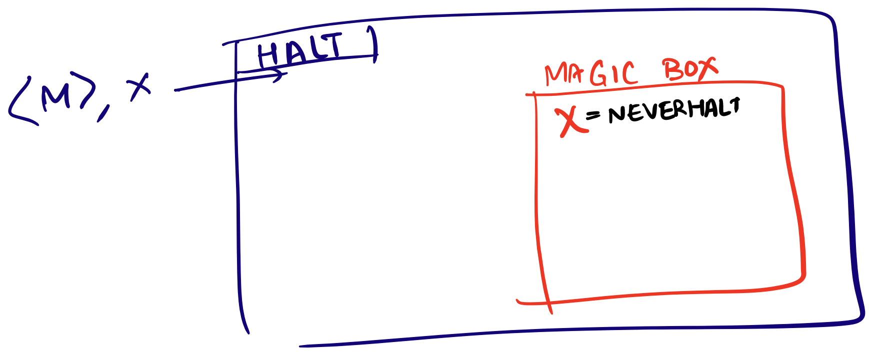
$L_{HALT} = \{ \langle M \rangle, x \mid M$ on input x halts $\}$.



L_{HALT} is the canonical undecidable language.

L_{HALT} is the circuitSAT of undecidability.

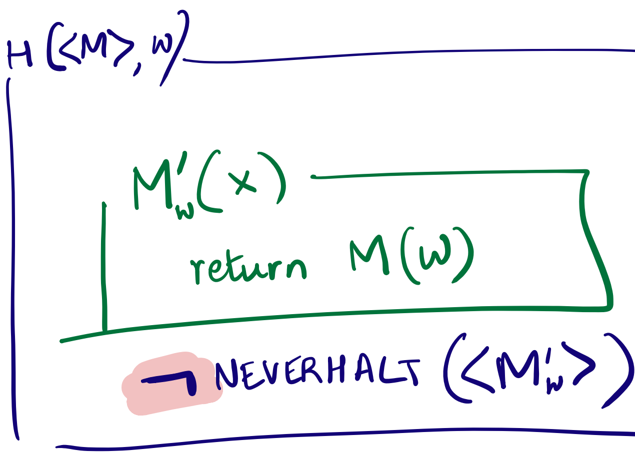
When trying to prove that problem X is undecidable, reduce FROM HALT TO X.



X = NEVERHALT problem.

Given ⟨M⟩, does M always halt (on every input)?

If it does, NEVERHALT(⟨M⟩) says NO, otherwise YES.



Case 1 → M halts on input w.

⇒ M_w' halts on every input x.

⇒ NEVERHALT(⟨M_w'⟩) returns NO.

⇒ H(⟨M⟩, w) should return YES.

Case 2 → M does not halt on input w.

⇒ M_w' does not halt on every input x.

⇒ NEVERHALT(⟨M_w'⟩) returns YES

⇒ H(⟨M⟩, w) returns NO. $\Rightarrow\!\Leftarrow$

$L_{\text{"374"}} = \{ \langle M \rangle \mid M \text{ accepts the string "374"} \}.$

MAGIC BOX

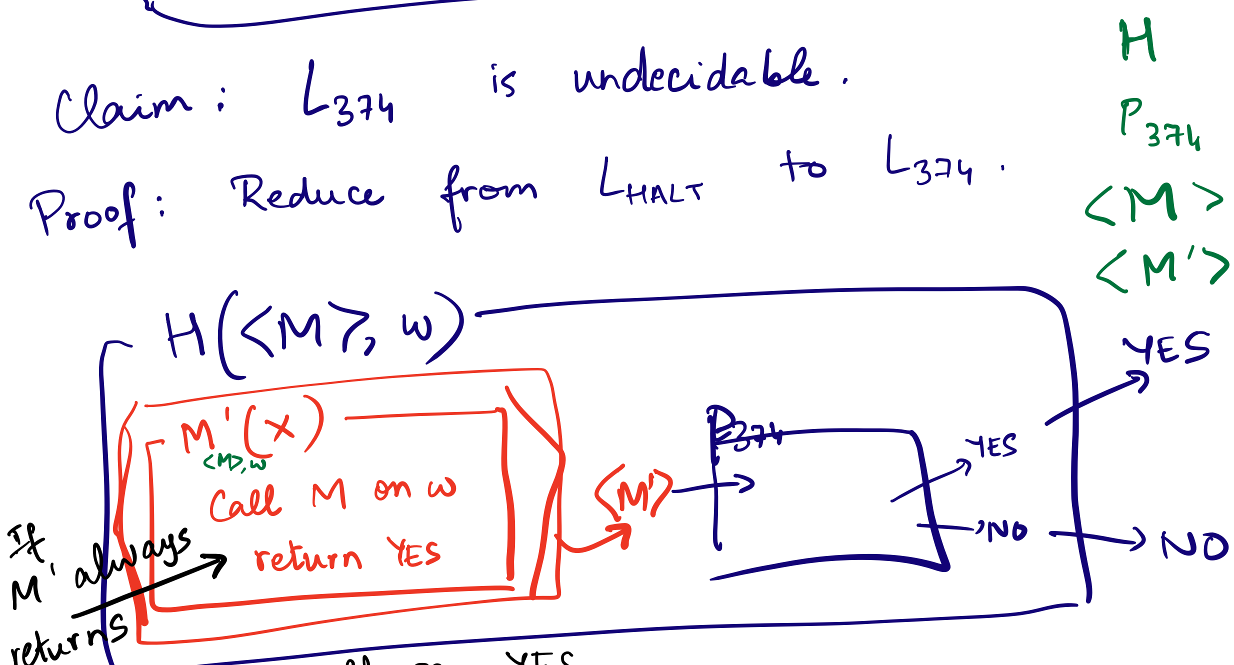
$P_{374}(\langle M \rangle)$

If M accepts 374 : $\boxed{\text{YES}}$

If M does not accept / loops : $\boxed{\text{NO}}$

Claim : L_{374} is undecidable.

Proof : Reduce from L_{HALT} to L_{374}.



$\Rightarrow$ P_{374} will say YES.

M halts on w

$\Rightarrow$ M' halts and returns YES on all inputs.

$\Rightarrow$ $P_{374}\langle M' \rangle$ says YES. $\Rightarrow$ H says YES.

M does not halt on w

$\Rightarrow$ M' does not halt on any input

$\Rightarrow$ $P_{374}\langle M' \rangle$ say NO. $\Rightarrow$ H says NO

RICE'S THEOREM.

// $\text{Accept}(M) = \{w \mid M \text{ accepts } w\}$

Use: Given $\langle M \rangle$, does M accept ____ ?

$\boxed{\text{THM.}}$

Let $\mathcal{L}$ be <u>any</u> <u>set</u> of languages such that:

property.

- There is a program Y s.t. <u>$\text{Accept}(Y)$</u> $\in \mathcal{L}$.
- There is a program N s.t. $\text{Accept}(N) \notin \mathcal{L}$.

Then, deciding if $\text{Accept}(M) \in \mathcal{L}$ (for all M)

is impossible.

(You want a program $P(\langle M \rangle)$ outputs 1

iff $\text{Accept}(M) \in \mathcal{L}$).

- Given $\langle M \rangle$, does M accept "374"?

  $\mathcal{L}$ = set of all langs. that contain string "374".

  Y: accept everything N: reject everything.

  Given $\langle M \rangle$, does:

strings

- M accept all palindromes ?

$\mathcal{L} = \{ \{\text{all palindromes}\}, \Sigma^*, \ldots\ldots \}$

Y: $\boxed{\text{accept everything}}$ $\in \mathcal{L}$ N: reject everything $\notin \mathcal{L}$.

- Language of M is non-regular ?

$\mathcal{L} = \{$ set of all non-regular languages $\}$.

Y: accept $\{0^n 1^n \mid n > 0\} \in \mathcal{L}$. and nothing else

N: accept every string. $\text{ACCEPT}(N) = \Sigma^* \notin \mathcal{L}$.

- M accepts only "374" ?

$\mathcal{L} = \{\{374\}\}$ Y: accepts 374, nothing else N: accept nothing.

- M accepts "374".

iff $\text{ACC}(M) \in \mathcal{L}$ where

$\mathcal{L} = \{ \{374\}, \{1, 374\}, \{2, 374\} \ldots\ldots \{\ldots 374 \ldots\} \}$.

Set of all strings $\in \mathcal{L}$

Y that accepts everything has $\text{Accept}(Y) \in \mathcal{L}$.

N that rejects everything has $\text{ACCEPT}(N) = \emptyset \notin \mathcal{L}$.