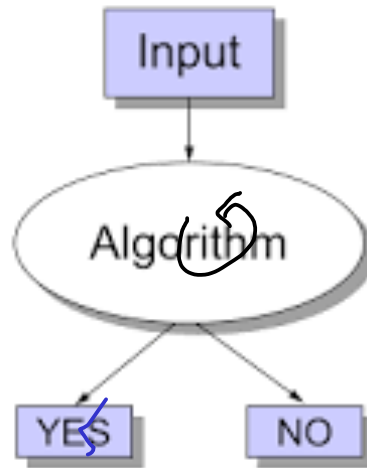


Problemas Decidíveis

Problemas de Decisão

- Problema de decisão é uma questão sobre um sistema formal com uma resposta do tipo sim-ou-não



$$f(x) = y$$

$$(f(x), y)$$

while(1);

Problemas de Decisão

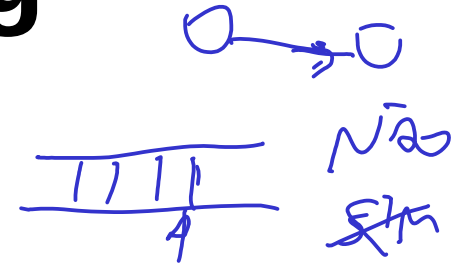
- Problemas decidíveis
 - Tem algoritmo que sempre para e responde sim ou não
- Problema não decidível
 - Não existe algoritmo que sempre pare

Problemas de Decisão

- Problema parcialmente decidível
 - Existe um algoritmo que resolve o problema quando a resposta é afirmativa.
 - Porém, quando a resposta é negativa, o algoritmo pode não parar.

Máquinas de Turing

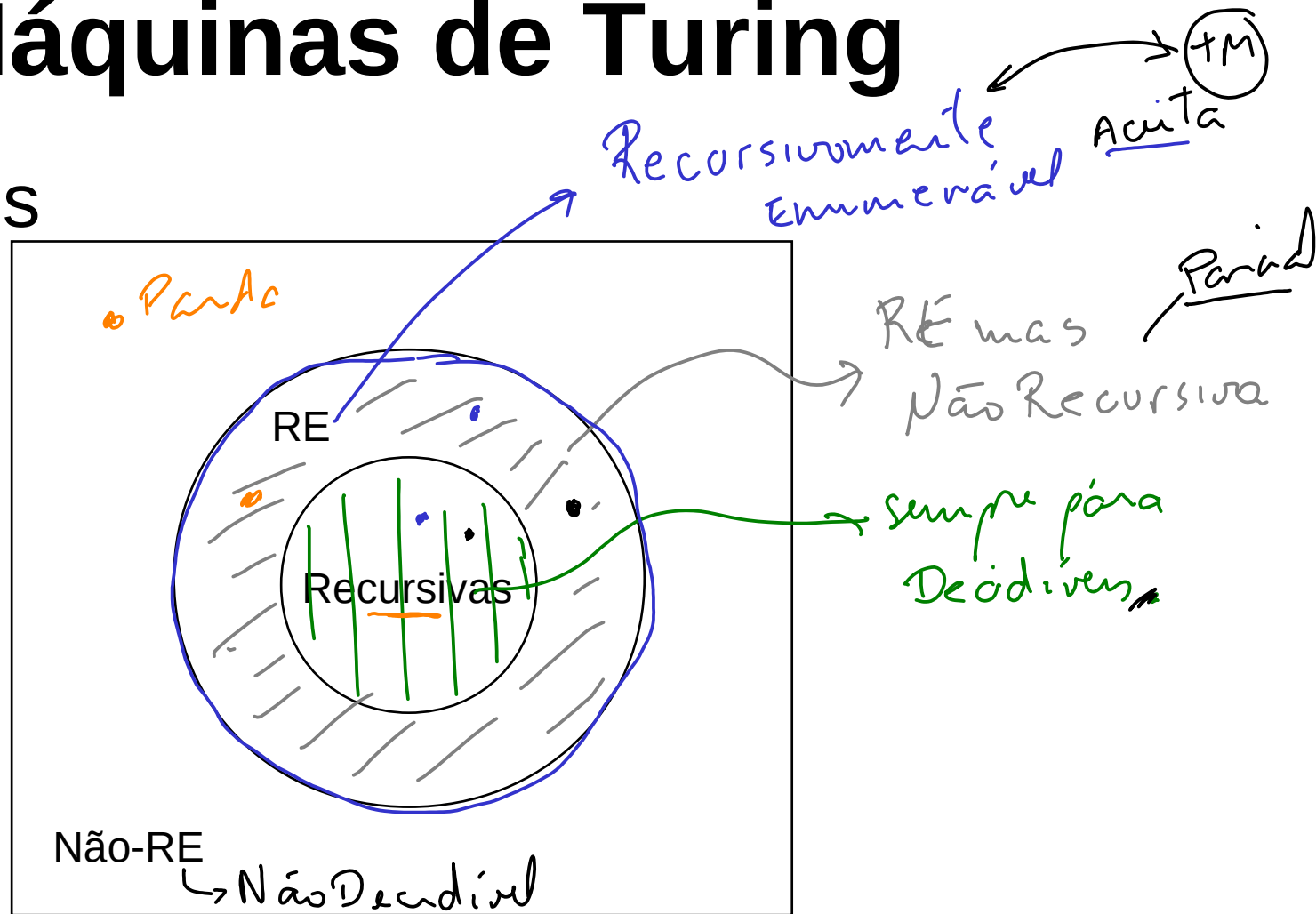
- Podem simular computadores reais
- Problemas que tem um algoritmo
 - Máquina de Turing pára, quer aceite ou não sua entrada
- Problemas que não tem um algoritmo
 - Máquinas de Turing podem funcionar indefinidamente, sobre entradas que não aceitam



Máquinas de Turing

- Linguagens

Redução



Linguagem RE

- Recursivamente Enumerável
 - Se $L = L(M)$ para alguma TM M
Reconhecida
 - Conjunto de linguagens que podemos aceitar ✓
usando uma máquina de Turing

Linguagem Recursiva

- Dizemos que L é recursiva se $L = L(M)$ para uma máquina de Turing M :
 - Se w está[✓] em L , então M a aceita E portanto pára
 - Se w não[✗] está em L , então M pára eventualmente, embora nunca entre em um estado de aceitação
- Noção informal de um “algoritmo” que sempre termina e produz uma resposta

Hipótese de Turing-Church

- estabelece uma correspondência entre as noções de Algoritmo e Máquina de Turing
- Problema Decidível \rightarrow Recursivo

Linguagem Não-RE

- Não Recursivamente Enumeráveis
 - Não podem ser representadas por Máquinas de Turing
 - Ou seja, não tem solução computacional

Problema da Parada

- "Dadas uma descrição de um programa e uma entrada finita, decida se o programa termina de rodar ou rodará indefinidamente."
- Alan Turing provou em 1936 que um algoritmo genérico para resolver o problema da parada para todos pares programa-entrada possíveis não pode existir.