

Decidibilidad

Máquinas de Turing

- Definição formal
 - Reconhecimento x transdução
- Generalizações e Restrições
 - Sem perda de representatividade
- Equivalência de modelos computacionais
 - MT na teoria e computador na prática

Tese de Turing-Church

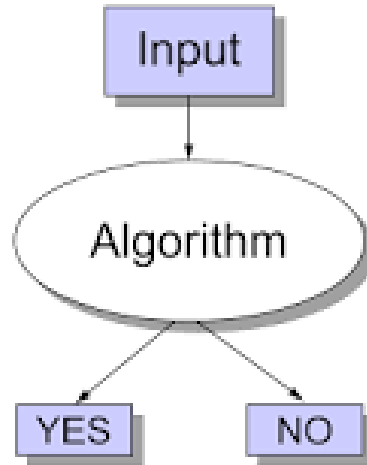
- Qualquer programa de computador pode ser traduzido em uma máquina de Turing
- E qualquer máquina de Turing pode ser traduzida para uma linguagem de programação de propósito geral

Tese de Turing-Church

- Assim, a tese é equivalente a dizer que qualquer linguagem de programação de propósito geral é suficiente para expressar qualquer algoritmo

Problemas de Decisão

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



Problema de Decisão

- Exemplos:
 - "dados dois números x e y , y é divisível por x ?"
 - "dado um número inteiro x , x é um número primo?"

Problema de Decisão

- Métodos usados para resolver problemas de decisão são chamados de procedimentos ou algoritmos
- Um problema de decisão que pode ser resolvido por algum algoritmo é chamado de problema de decisão **decidível**.

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

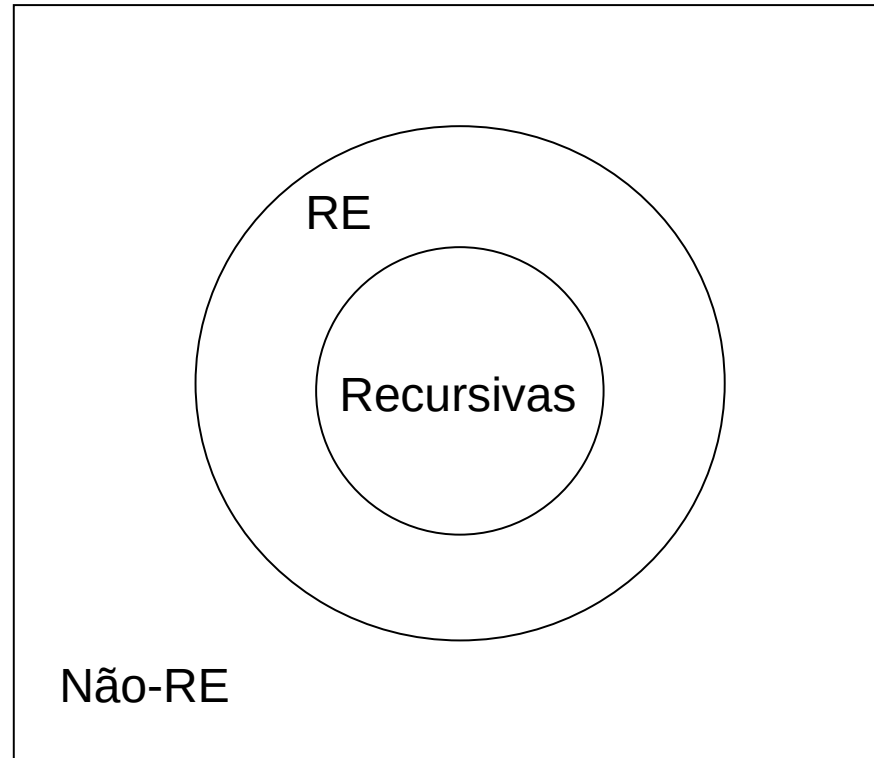
- Problema parcialmente decidível
 - Existe um algoritmo que resolve o problema quando a resposta é afirmativa.
 - Porém, quando a resposta é negativa, o algoritmo pode não parar.

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



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 para
 - 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

Linguagem RE

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

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.