



**Universidade Federal da Fronteira Sul**  
**Curso de Ciência da Computação**  
**Informática Básica**



# Máquina de Turing

Giancarlo D. Salton  
gian@uffs.edu.br

**Chapecó, Brasil**

# Introdução

## Alan Turing

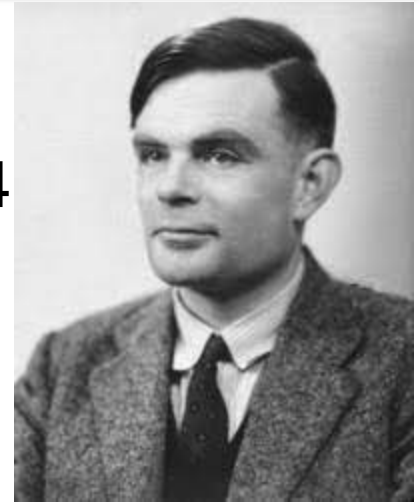
★ 23 de Junho de 1912 – † 7 de Junho de 1954

- Considerado o “pai” da computação
- Contribuições:

Máquina de Turing e a tese de Church-Turing

Colossus (Enigma)

Teste de Turing



# Introdução

## Decidibilidade

- o termo decidível se refere a um problema de decisão, ou seja, a questão da existência de um método efetivo para determinar a pertinência em um conjunto de fórmulas

## Problemas decidíveis

- um sistema lógico fixo é decidível se existe um algoritmo eficiente para determinar se fórmulas arbitrárias pertencem a ela

## Problemas indecidíveis

Não existe algoritmo

## Computabilidade

Os algoritmos conhecidos são demasiado dispendiosos

Simplificação; heurísticas

# Introdução

## Decidibilidade

### David Hilbert (início do séc. XX)

- Há alguma maneira de determinar se qualquer fórmula da lógica de predicados de primeira ordem, aplicada aos inteiros, é verdadeira?

### Kurt Gödel (1931)

- Teorema da incompletude: construiu uma fórmula que não pode ser provada nem refutada

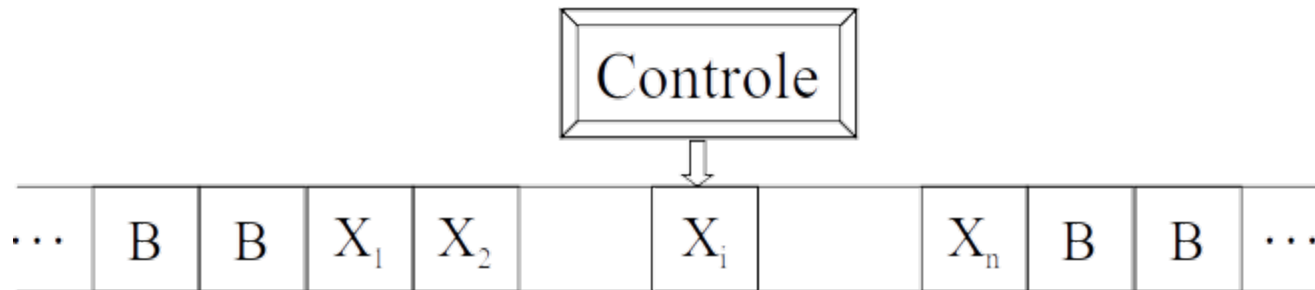
### Alan Turing (1936)

- Máquina de Turing: modelo de qualquer computação possível

### Hipótese de Church (tese de Church-Turing, não demonstrável)

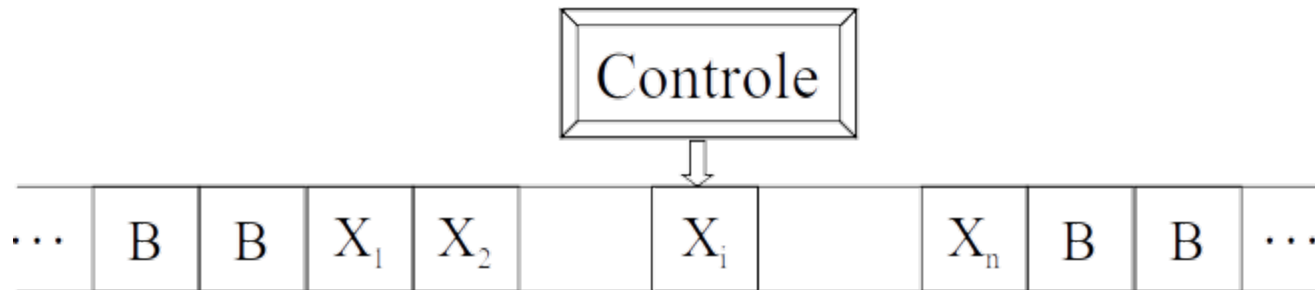
- Todos os modelos gerais de computação são equivalentes às funções parciais recursivas e às máquinas de Turing (mesmo os computadores atuais)

# Máquina de Turing



- Fita de comprimento infinito dividida em células
  - Cada célula pode conter um símbolo (alfabeto finito)
- Controle finito
  - Número finito de estados
- Entrada
  - Cadeia finita constituída por símbolos do alfabeto de entrada
- Cabeça da fita
  - Sempre posicionada em uma célula
  - No início, está na célula mais à esquerda da entrada

# Máquina de Turing



- Baseado no estado atual e no valor lido realiza o movimento
- Movimento ou passo da máquina

Função do estado e do símbolo a ser lido pela cabeça

1. Mudança de estado

pode permanecer no mesmo

2. Escrita de um símbolo na célula onde está a cabeça

pode ser o mesmo

3. Deslocamento da cabeça uma célula à esquerda ou à direita

# Máquina de Turing

- Máquina de Turing (TM)  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$

$Q$ : conjunto finito de estados de controle

$\Sigma$ : conjunto finito de símbolos de entrada

$\Gamma$ : conjunto finito de símbolos da fita

$\delta$ : função de transição  $(q, X) = (p, Y, D)$

$q$  é um estado,  $X$  um símbolo da fita

$p$  é o novo estado, em  $Q$ ;

$Y$  é o símbolo que substitui  $X$ ;

$D$  é L ou R, esquerda ou direita, direção em que a cabeça se move depois da substituição

$q_0$ : estado inicial

$B$ : branco, símbolo que preenche a fita, exceto as células com a entrada

$F$ : conjunto de estados de aceitação ou finais

# Máquina de Turing

Exemplo: Implementação de uma máquina de Turing que verifica se um número binário é par ou não.

Um número binário, quando par, tem seu bit menos significativo igual a 0, e quando ímpar igual a 1.

Á máquina percorre a fita até encontrar um símbolo em branco “ $\sqcup$ ”, então basta ela voltar uma posição e verificar a paridade. Se for 0 ela retorna  $q\_ok$ , se for 1 ela retorna  $q\_fim$ . Segue o conjunto de estados ( $Q$ ) e as funções de transições da máquina ( $\delta$ ) (programa da máquina):

$$Q = \{q\_inicio, q\_teste, q\_ok, q\_fim\}$$

$$\Sigma = \{0, 1, \sqcup\}$$

$$q_0 = q\_inicio$$

$$F = \{q\_ok, q\_fim\}$$



# Máquina de Turing

$\delta =$

**q\_inicio (#1)**

(q, s)	$\delta(s, q)$
(q_inicio, 0)	(0, q_inicio, ->)
(q_inicio, 1)	(1, q_inicio, ->)
(q_inicio, $\sqcup$ )	( $\sqcup$ , q_teste, <-)

**q\_teste (#2)**

	$\delta(s, q)$
(q_teste, 0)	(0, q_ok, -)
(q_teste, 1)	(1, q_fim, -)
(q_teste, $\sqcup$ )	don't care

# Máquina de Turing

$\delta =$

**q\_inicio (#1)**

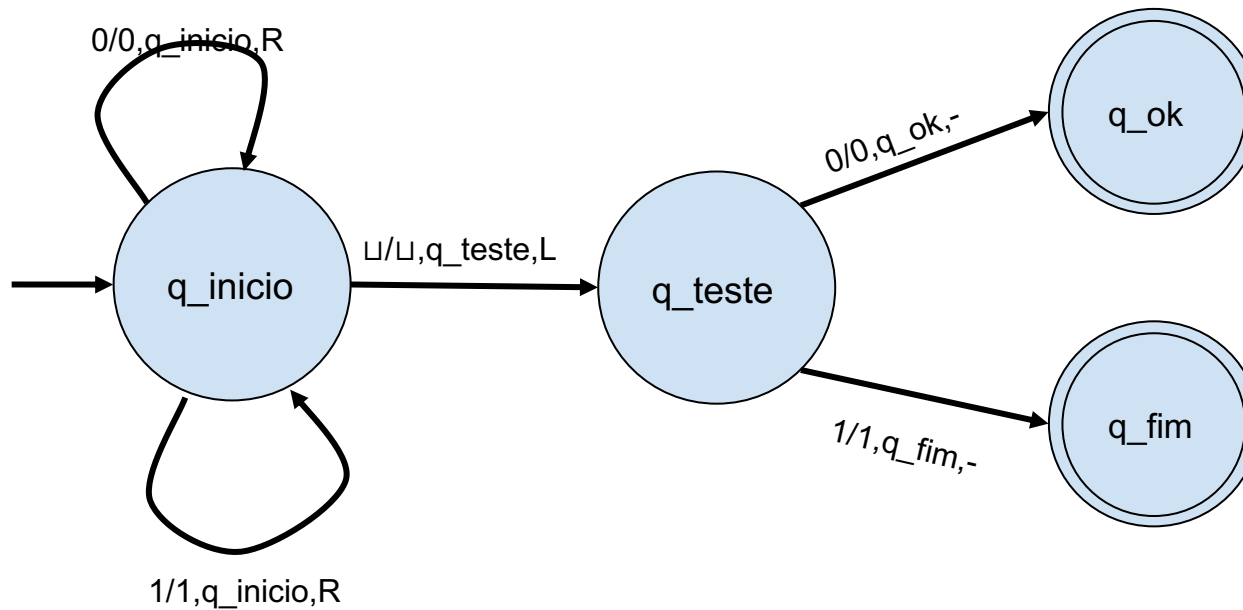
(q, s)	$\delta(s, q)$
(q_inicio, 0)	(0, q_inicio, ->)
(q_inicio, 1)	(1, q_inicio, ->)
(q_inicio, $\sqcup$ )	( $\sqcup$ , q_teste, <-)

**q\_teste (#2)**

	$\delta(s, q)$
(q_teste, 0)	(0, q_ok, -)
(q_teste, 1)	(1, q_fim, -)
(q_teste, $\sqcup$ )	don't care

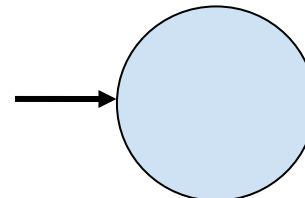
<https://turingmachinesimulator.com/shared/snzvukpgzb>

# Máquina de Turing

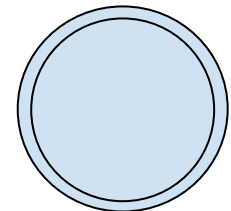


Representações:

estado inicial



estado final



# Máquina de Turing

Exercício:

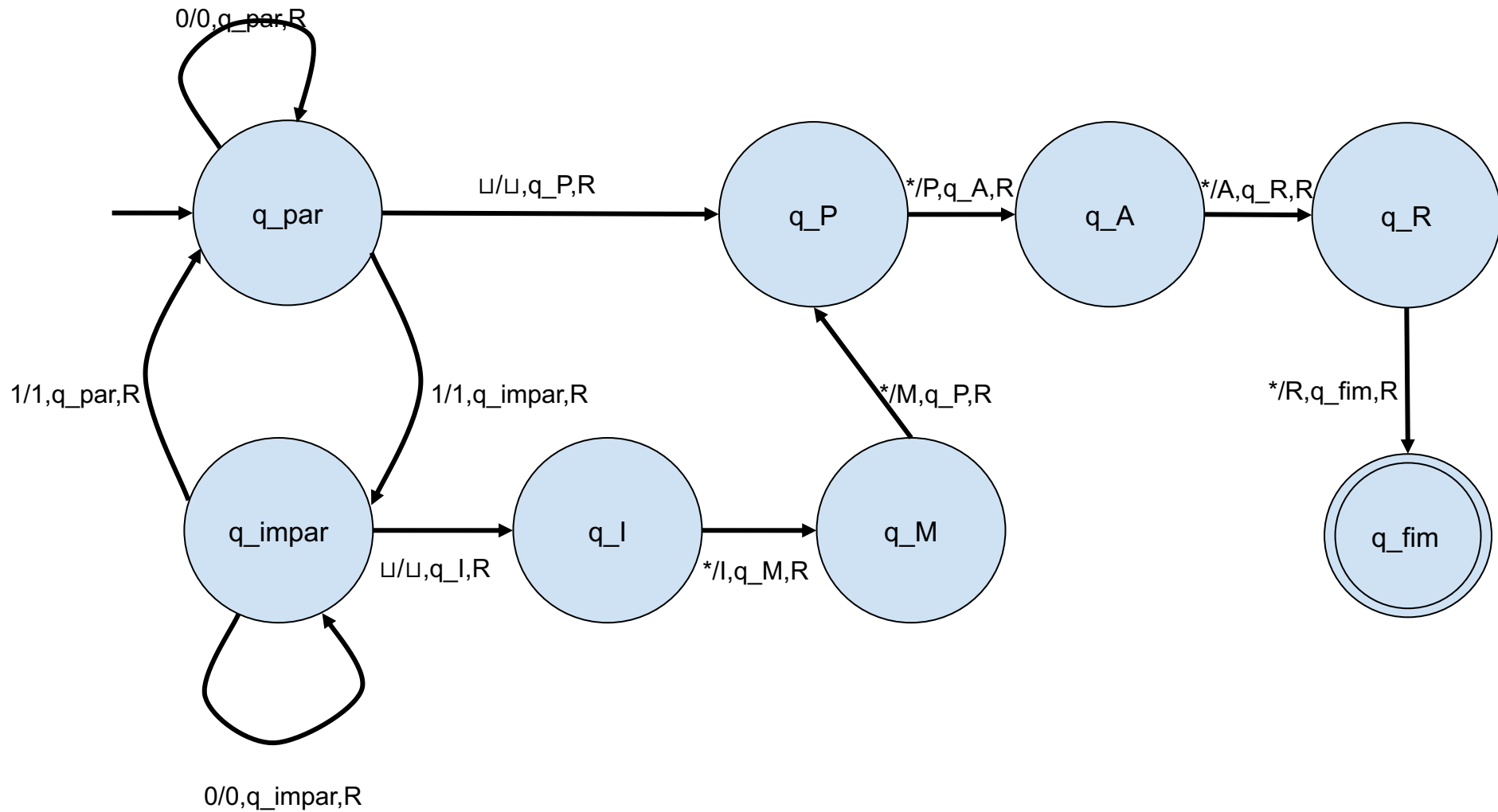
Implementar uma MT que verifica a paridade de um número.

A paridade ímpar é quando o número de '1's presentes no número é ímpar. A paridade é par quando a quantidade de '1's presentes no número é par. Considere o espaço( $\sqcup$ ) a indicação do fim do número.

Quando encontrar o espaço a MT deve escrever na fita a palavra PAR ou a palavra IMPAR conforme a paridade verificada

Apresente o conjunto de estados ( $Q$ ), o alfabeto ( $\Sigma$ ), as funções de transições ( $\delta$ ), o estado inicial ( $q_0$ ) e os estados finais ( $F$ )

# Máquina de Turing



# Máquina de Turing

## Exercício para entregar 1:

Implementação de uma Máquina de Turing para verificar um padrão de caracteres em uma fita. O exercício consiste em encontrar o padrão: "aab" em uma fita de caracteres aleatórios.

A máquina começa no estado  $q_{\text{inicio}}$  e deve percorrer a fita a direita até que encontre o padrão definido e então retornar  $q_{\text{ok}}$  ou até encontrar o fim da fita.

$$Q = \{q_{\text{inicio}}, q_{\text{ok}}, \dots\}$$

$$\Sigma = \{a, b, \sqcup\}$$

$$q_0 = q_{\text{inicio}}$$

$$F = \{q_{\text{ok}}, q_{\text{nao}}\}$$

Apresente: (a) o conjunto de estados completo (Q); (b) as funções de transições ( $\delta$ ) e; (c) o diagrama de transição de estados (d) o link para a solução no simulador (<https://turingmachinesimulator.com>) para o problema acima.

# Máquina de Turing

## Exercício para entregar 2:

Implementação de uma máquina de Turing para copiar a primeira sequência de caracteres “a”s encontrada na fita. A entrada está armazenada no lado direito do cabeçote e a cópia da sequência é feita no lado esquerdo. A máquina percorre a fita até encontrar um caractere “a” e então inicia a cópia dos caracteres “a” encontrados em sequência. Após o fim da cópia a máquina vai para o estado  $q_{ok}$ , se a máquina não encontrar nenhum caractere “a” ela vai para o estado  $q_{fim}$ .

$$Q = \{q_{ok}, q_{fim}, \dots\}$$

$$\Sigma = \{a, b, \sqcup, \dots\}$$

$$q_0 = ?$$

$$F = \{q_{ok}\}$$

Apresente: (a) o conjunto de estados completo ( $Q$ ); (b) as funções de transições ( $\delta$ ) e; (c) o diagrama de transição de estados (d) o link para a solução no simulador (<https://turingmachinesimulator.com>) para o problema acima.