



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Máquina de estado finito: pode representar um modelo computacional.

Máquina de Turing: Máquina de estado que representa um modelo computacional com memória adicional.



Introdução à Informática

Máquinas de estado II: Máquinas de Turing



Alan Mathison Turing propôs o modelo das máquinas de estado denominadas *Logical Computer Machines* (LCM) que hoje conhecemos como Máquinas de Turing.

Sua tese provou a computabilidade ou seja, como um sistema formal podia executar operações automáticas com auxílio de um dispositivo de memória.



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Automação de operações: manipulação de símbolos e um conjunto de regras.

Máquina de Turing: como um dispositivo semelhante à uma máquina de estado finito, com a habilidade adicional de ler dados de entrada e, ainda, apagar e regravar os dados de entrada como uma memória ilimitada.

O fato de a memória de uma máquina de Turing ser ilimitada faz com que ela seja uma ``máquina ideal'' (um modelo) e não uma ``máquina real''.



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Uma máquina de Turing é definida por 4 funções declaradas em um conjunto de 5 elementos de entrada. Estas funções são:

- Pára (não faz nada);
- Imprime um símbolo do alfabeto definido na posição lida (pode ser o mesmo);
- Passa para o próximo estado (pode ser o mesmo);
- Move a cabeça de leitura para a direita ou esquerda, de uma posição.



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Os 5 elementos de entrada formam uma quintupla (e,s,i,p,d) que representam:

e	s	i	p	d
↓	↓	↓	↓	↓
estado atual	símbolo atual	símbolo impresso	próximo estado	direção do movimento





Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Exemplo – Considere uma máquina de Turing descrita pelas seguintes quintuplas:

$(0,0,1,0,D)$

$(0,1,0,0,D)$

$(0,b,1,1,E)$

$(1,0,0,1,D)$

$(1,1,0,1,D)$



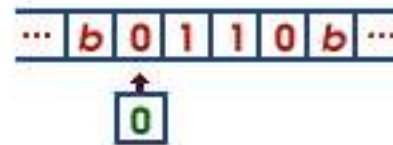
Introdução à Informática

Máquinas de estado II: Máquinas de Turing

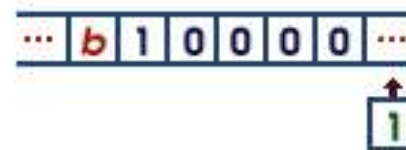
Neste exemplo:

(0,0,1,0,D)
(0,1,0,0,D)
(0,b,1,1,E)
(1,0,0,1,D)
(1,1,0,1,D)

Considere o conjunto inicial de símbolos:



Resultado:



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Considere a seguinte máquina de Turing:

(0,0,0,1,D)
(0,1,0,0,D)
(0,b,b,0,D)
(1,0,1,0,D)
(1,1,1,0,E)

a) Descreva a última fita se a fita inicial é:

| b | 1 | 0 | b |
↑
| 0 |

b) Descreva o comportamento da máquina se a fita for:

| b | 0 | 1 | b |
↑
| 0 |



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Uma máquina de Turing pode ser usada para reconhecer uma seqüência:

Uma máquina de Turing T pode ser usada para reconhecer o um subconjunto S de uma fita I , contendo uma cadeia C de caracteres se, e somente se, $C \in S$.

Imagine uma máquina de Turing que reconheça:

$$S = \{ 0^n 1^n, n \geq 0 \}$$

ou seja, qualquer conjunto $\{0...01...1\}$ com número igual de zeros e uns.



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

(0,b,b,8,D): reconhece a fita vazia
(0,0,#,1,D): apaga o "0" mais à esquerda
(1,0,0,1,D): move à direita pelos símbolos "0"
(1,1,1,1,D): move à direita pelos símbolos "1"
(1,b,b,2,E): encontrou o fim da seqüência
(2,1,#,3,E): apaga o "1" mais à direita
(3,1,1,3,E): move à esquerda pelos símbolos "1"
(3,0,0,4,E): passa para o estado 4 se existir zeros
(3,#,#,7,D): passa para o estado 7 se não existir "0"
(4,0,0,4,E): move à esquerda pelos símbolos "0"
(4,#,#,5,D): acha a extremidade esquerda e move-se para a direita
(5,0,#,6,D): apaga mais um "0" à esquerda e move-se para a direita
(6,0,0,6,D): move-se para a direita no estado 6
(6,1,1,6,D): até o final da seqüência e, então
(6,#,#,2,E): move-se para a esquerda no estado 2
(7,#,#,8,D): não tem mais símbolos "1"; muda para o estado 8 e pára.



A máquina só reconhece a seqüência se pára no estado 8.

Introdução à Informática

Máquinas de estado II: Máquinas de Turing

(0,b,b,8,D): reconhece a fita vazia
(0,0,#,1,D): apaga o "0" mais à esquerda
(1,0,0,1,D): move à direita pelos símbolos "0"
(1,1,1,1,D): move à direita pelos símbolos "1"
(1,b,b,2,E): encontrou o fim da seqüência
(2,1,#,3,E): apaga o "1" mais à direita
(3,1,1,3,E): move à esquerda pelos símbolos "1"
(3,0,0,4,E): passa para o estado 4 se existir zeros
(3,#,#,7,D): passa para o estado 7 se não existir "0"
(4,0,0,4,E): move à esquerda pelos símbolos "0"
(4,#,#,5,D): acha a extremidade esquerda e move-se para a direita
(5,0,#,6,D): apaga mais um "0" à esquerda e move-se para a direita
(6,0,0,6,D): move-se para a direita no estado 6
(6,1,1,6,D): até o final da seqüência e, então
(6,#,#,2,E): move-se para a esquerda no estado 2
(7,#,#,8,D): não tem mais símbolos "1"; muda para o estado 8 e pára.



Verifique esta máquina!



Introdução à Informática

Máquinas de estado II: Máquinas de Turing

Exercício em sala:

Projete uma máquina de Turing que troque, em uma fita contendo qualquer conjunto de $\{0,1\}$, todos os símbolos 0 por 1 e vice-versa.

