

Máquinas de Turing

Computabilidade por Máquinas de Turing

Prof. Edson Alves

Faculdade UnB Gama

2020

Sumário

1. Definição de computabilidade
2. Exemplos de Máquinas de Turing

Funções efetivamente computáveis

Definição

Uma função f de inteiros positivos em inteiros positivos é **efetivamente computável** se existe uma lista de instrução que, em princípio, permitam computar $f(n)$ para qualquer argumento n .

Observações: a noção de computabilidade efetiva é intuitiva, pois não é rigorosamente definida. Contudo, as instruções da lista citada devem ser definidas e explícitas, de tal modo que possam ser seguidas sem exigir informações externas ou engenhosidade para a sua execução.

Máquina de Turing

Definição

Uma **máquina de Turing** é uma máquina idealizada para realizar computação em números inteiros positivos usando notação monádica (onde o inteiro positivo n é representado por n traços). A computação acontece em uma fita linear, dividida em quadrados, infinita em ambas direções (esquerda e direita). Cada quadrado ou está **em branco** (representado pelos símbolos $S_0, 0$ ou B) ou tem **um traço** ($S_1, 1$ ou |). Exceto por um número finito de exceções, todos os demais quadrados estão em branco.

Observações: cada etapa da computação acontece em um quadrado da fita. O computador (agente humano, mecânico ou eletrônico) pode apagar o traço, caso o quadrado contenha um, ou escrever um traço, caso o quadrado esteja vazio. Além disso, ele pode mover a cabeça para o quadrado à esquerda, ou para o quadrado da direita.

Resumo das instruções possíveis para uma máquina de Turing

As instruções tem forma condicional, dizendo o que fazer caso o quadrado esteja em branco (S_0) ou contenha um traço (S_1):

- (1) **Apagar:** escrever S_0 no quadrado, independente de seu estado
- (2) **Escrever:** escrever S_1 no quadrado, independente de seu estado
- (3) **Mover** para à esquerda (R)
- (4) **Mover** para à direita (L)
- (5) **Parar** a computação

A instrução (1) em um quadrado em branco, ou a instrução (2) em um quadrado com um traço, equivalem a não fazer nada.

Estados e programa

- ▶ Em cada etapa da computação, o computador avalia um quadrado em particular da fita
- ▶ O **estado** atual da máquina determina qual instrução (ação) a ser realizada e qual será o próximo estado que a máquina assumirá, a depender se há um traço ou não no quadrado em avaliação
- ▶ Assim, cada etapa da computação depende do estado atual e o símbolo contido no quadrado a ser avaliado
- ▶ Em cada etapa é realizada uma das cinco instruções listadas anteriormente, e é determinado o próximo estado que a máquina assumirá
- ▶ Um **programa** consiste na descrição de todos os estados possíveis da máquina, e de todas as ações a serem seguidas em cada estado, a depender dos símbolos encontrados no quadrado a ser avaliado

Tabela de Máquina

Definição

Uma **tabela de máquina** é uma tabela bidimensional cujas linhas representam os possíveis estados da máquina de Turing, e as duas colunas representam os símbolos que podem estar escritos no quadrado a ser avaliado (S_0 ou S_1). Cada célula descreve a ação a ser realizada, a depender do símbolo escrito, seguida do estado que sucederá o estado atual.

Em uma fita inicialmente como todos os quadrados em branco, o programa abaixo escreve três traços consecutivos, e para.

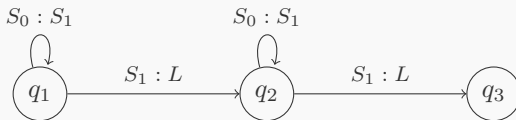
	S_0	S_1
q_1	$S_1 q_1$	$L q_2$
q_2	$S_1 q_2$	$L q_3$
q_3	$S_1 q_3$	

Fluxograma

Fluxograma

Em um programa escrito em **fluxograma** cada estado possível da máquina é representado por um círculo, e as transições possíveis são representadas por setas que partem do estado atual para o próximo estado, com rótulos na forma $S_i : I_k$, onde S_i é o símbolo presente no quadrado a ser avaliado e I_k é a instrução a ser seguida.

Mesmo programa apresentado na tabela de máquina. A menos que indicado de outra maneira, é assumido que a máquina inicia a computação no estado de menor número.



Conjunto de Quádruplas

Definição

Um programa pode ser descrito por um conjunto de **quádruplas** (q_a, S_i, I_k, q_b) , onde q_a é o estado atual, S_i o símbolo escrito no quadrado a ser avaliado, I_k é a instrução (ação) a ser seguida e q_b é o próximo estado. Se não houver ambiguidade, a quádrupla pode ser notada sem parêntesis e vírgulas, isto é, $q_a S_i I_k q_b$.

O programa ilustrado anteriormente por meio de tabela de máquina e de fluxograma, em lista de quádruplas:

$$q_1 S_0 S_1 q_1, \quad q_1 S_1 L q_2, \quad q_2 S_0 S_1 q_2, \quad q_2 S_1 L q_3, \quad q_3 S_0 S_1 q_3$$

Configurações

- ▶ O funcionamento de uma máquina de Turing pode ser descrito por meio de uma sequência de **configurações**
- ▶ Cada configuração lista todos os símbolos escritos na fita, o estado atual e o quadrado a ser examinado
- ▶ Por exemplo, a configuração

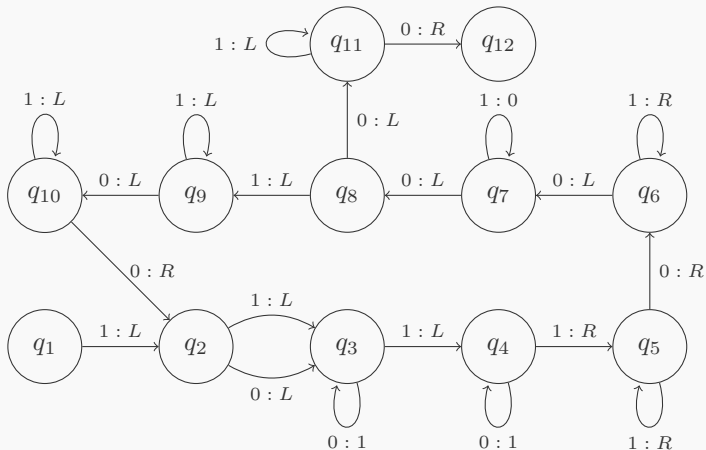
11010₃101

representa uma fita com dois traços, um espaço em branco, um novo traço, um quadrado em branco, que está sendo examinado pelo estado 3, um traço, um espaço em branco e um traço

- ▶ Assume-se que os quadrados não listados contém, todos, espaços em branco
- ▶ Assim, o estado descrito acima seria idêntico aos estados 011010₃101 e 11010₃10100, por exemplo

Exemplo: Duplicando o número de traços

Assuma que a máquina abaixo começa no traço mais à esquerda de um bloco de n traços de uma fita que, de resto, está em branco. Esta máquina para no traço mais à esquerda de um bloco de $2n$ traços de uma fita que, de resto, está em branco.



Configurações para uma máquina com $n = 2$

- ▶ A configuração inicial da máquina é

$$1_11$$

- ▶ O estado 1 move o computador para à esquerda, levando à configuração

$$0_211$$

- ▶ Como o computador está avaliando um quadrado vazio, as próximas configurações são

$$0_3011, \quad 1_3011, \quad 0_41011, \quad 1_41011$$

- ▶ Esta sequência de instruções criou um par de traços à esquerda do bloco original, separado deste por um espaço em branco

Configurações para uma máquina com $n = 2$

- ▶ Os dois próximos estados (5 e 6) movimentam o computador para o último traço do segundo bloco (bloco original):

11_5011 , 110_511 , 1101_61 , 11011_6 , 110110_6 , 11011_7

- ▶ O estado 7 apaga o último traço do bloco (se existir), e em seguida segue para à esquerda

11010_7 , 1101_8

- ▶ Como há ainda um traço no bloco original, o estado 8 segue para o 9, que salta o bloco original, e em seguida para o 10, que posiciona a máquina no último traço do bloco à esquerda:

110_91 , $11_{10}01$, $1_{10}101$, $0_{10}1101$

Configurações para uma máquina com $n = 2$

- ▶ De volta ao estado 2, a máquina segue para escrever mais um par de traços à esquerda do bloco à esquerda:

$1_2101, 0_31101, 1_31101, 0_411101, 1_411101$

- ▶ Novamente os estados 5 e 6 posicionarão o computador no último traço do bloco original:

$11_51101, 111_5101, 1111_501, 11110_51, 111101_6, 1111010_6$

- ▶ O estado 7 apaga o último traço restante, e segue para o estado 8:

$111101_7, 111100_7, 11110_8$

- ▶ Como não há mais traços no bloco original, o estado 8 vai para o estado 11, o qual irá posicionar o computador no traço mais à esquerda do bloco restante:

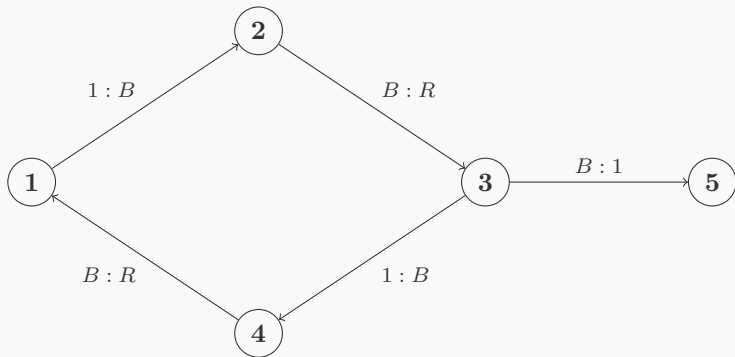
$1111_{11}, 111_{11}1, 11_{11}11, 1_{11}111, 0_{11}1111, 1_{12}111$

Exemplo: Paridade

A máquina abaixo inicia no traço mais à esquerda de um bloco de n traços de uma fita que, de resto, está em branco, e termina em um quadrado de uma fita, que de resto está em branco, que contém um traço, se n é ímpar, ou está em branco, se n é par.

Exemplo: Paridade

A máquina abaixo inicia no traço mais à esquerda de um bloco de n traços de uma fita que, de resto, está em branco, e termina em um quadrado de uma fita, que de resto está em branco, que contém um traço, se n é ímpar, ou está em branco, se n é par.



Referências

1. **BOOLOS**, George S.; **BURGESS**, John P.; **JEFFREY**, Richard C. *Computabilidade e Lógica*, Editora Unesp, 2012.