Aula 11: Linguagens Recursivamente Enumeráveis e Sensíveis ao Contexto.

Prof. Lucio A. Rocha

Engenharia de Computação
Universidade Tecnológica Federal do Paraná, UTFPR
Campus Apucarana, Brasil

1º semestre / 2023

Sumário

1. Linguagens Recursivamente Enumeráveis
2. Linguagens Sensíveis ao Contexto

Seção 1

Linguagens Recursivamente Enumeráveis

Linguagens Recursivamente Enumeráveis

- Motivação: implementar qualquer função computável.
- Função computável: é aquela para a qual existe um algoritmo que calcula os valores de saída a partir dos valores de entrada.
  - Problemas computáveis:
    - $L = \{w\#w | w \in \{0,1\}^*\}$
    - Sucessor de um número em $\mathbb{N}$, $f(x) = x^2$, sequência de Fibonnaci, fatorial(x), etc.

Linguagens Recursivamente Enumeráveis

- Máquina de Turing (MT): é a formalização de uma função computável.
- Também é a formalização de um algoritmo.
- Algoritmo:
  - Sequência ordenada, mecânica e não-ambígua de instruções finitas.

Linguagens Recursivamente Enumeráveis

- MT: executa algoritmos.
  - Utiliza uma fita infinita que é a sua memória.
  - Cabeçote para ler e escrever símbolos e mover o cabeçote sobre a fita.
- Inicialmente, a fita contém apenas a palavra de entrada.
  - Se a máquina precisa armazenar informação, ela pode escrever na fita.
  - Para ler a informação que foi escrita, a máquina pode mover o cabeçote sobre ela.
  - Uma única saída *ACEITA* ou *REJEITA* é obtida entrando em algum estado de aceitação ou rejeição.
  - Se não entra em estado de aceitação ou rejeição, a MT executa para sempre, e nunca pára.

Linguagens Recursivamente Enumeráveis

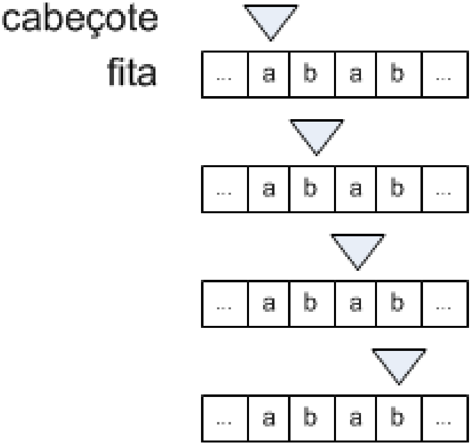

Figura: Representação de uma MT.

Linguagens Recursivamente Enumeráveis

- $1, x \rightarrow L$: Leu o símbolo 1 ou x, não escreve, vá para a esquerda (left).

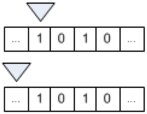

Figura: Leitura da fita com uma MT.

- $1 \rightarrow x, R$: Leu o símbolo 1, escreve x, vá para a direita (right).

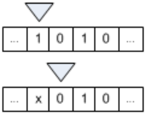

Figura: Escrita na fita com uma uma MT.

Linguagens Recursivamente Enumeráveis

- Diferença entre MT e AF:
  1. MT pode ler e escrever na fita.
  2. A leitura/escrita pode mover o cabeçote para a esquerda ou para a direita.
  3. A fita é infinita.
  4. Estados especiais para aceitar / rejeitar tomam efeito imediatamente.

Linguagens Recursivamente Enumeráveis

- Def.: Uma MT é uma 7-tupla:

$$MT = (\Sigma, Q, T, \delta, q_0, q_{aceita}, q_{rejeita})$$

onde Σ, Q, T são todos conjuntos finitos e:

  - Σ: é o alfabeto de entrada (não contém o símbolo vazio)
  - Q: é o conjunto de estados.
  - T: é o alfabeto da fita $| \ \varepsilon \in T$ e $\Sigma \subseteq T$
  - δ: é a função de transição (função parcial):

$$Q \times T \rightarrow Q \times T \times \{L, R\}$$

  - $q_0 \in Q$: é o estado inicial.
  - $q_{aceita} \in Q$: é o estado *ACEITA*.
  - $q_{rejetia} \in Q$: é o estado *REJEITA*.

Linguagens Recursivamente Enumeráveis

- Em uma MT, o símbolo vazio indica o fim da fita.
- Se o cabeçote está no final da fita à direita e a transição pede para ir para a direita, o cabeçote fica onde está.
- A computação continua até que ela entre no estado de ACEITA ou REJEITA, e a computação pára.
- Caso contrário, a MT continua sempre.

Linguagens Recursivamente Enumeráveis

- Configuração de uma MT:

$$(u,q,v)$$

  - uv: conteúdo atual da fita
  - q: estado atual
  - posição do cabeçote: primeiro símbolo da string v
    - Exemplo: $1011q_7\underline{0}1111$
    - uv: $1011\underline{0}1111$
    - q_7: estado atual
    - posição do cabeçote: sobre o primeiro $\underline{0}$ de v.

Linguagens Recursivamente Enumeráveis

- A base de funcionamento de uma MT é a função de transição δ.
- A função de transição δ diz como a MT vai de um estado para o próximo estado.

$$\delta : Q \times T \rightarrow Q \times T \times \{L, R\}$$

- Exemplo: $\delta(q_0, a) = (q_1, b, L)$
  - Lê-se: dado o estado $q_0 \in Q$ e um símbolo $a \in T$ então:
    - escreva b (ação)
    - vá para o estado q_1 (próximo estado)
    - e mova o cabeçote para a esquerda (Left)
- Notação alternativa: $a \rightarrow b, L$

Linguagens Recursivamente Enumeráveis

- Uma MT M_1 aceita a entrada w se uma sequência de configurações $C_1, C_2, ..., C_k$ existe, tal que:
  1. C_1 é a configuração inicial de M_1 sobre a entrada w
  2. Cada C_i produz C_{i+1}
  3. C_k é uma configuração de aceitação
- $L(M_1)$ = coleção de palavras que M_1 aceita.

Linguagens Recursivamente Enumeráveis

Linguagem Recursivamente Enumerável

Def.: Uma linguagem é:

- Tipo 0,
- Recursivamente enumerável,
- Turing-reconhecível,

se existe uma Máquina de Turing que reconhece a linguagem, ou seja, verifica se a palavra pertence à linguagem.

- Uma máquina Turing-reconhecível pode aceitar ou rejeitar entradas, mas nem sempre pára para entradas que não pertecem à linguagem (pode entrar em loop).
- Se $w \notin L$, a MT pode parar ou entrar em loop.

Linguagens Recursivamente Enumeráveis

Linguagem Recursiva

Def.: Uma linguagem é:

- Recursiva,
- Decidível,
- Turing-decidível,

se existe, no mínimo, uma Máquina de Turing que decide a linguagem, ou seja, pára em um estado de *ACEITA* ou *REJEITA* para todas as entradas da linguagem.

- Uma máquina Turing-decidível pode aceitar ou rejeitar entradas, e <u>sempre</u> pára para todas as entradas.

Linguagens Recursivamente Enumeráveis

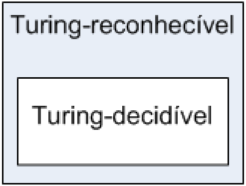

Figura: Linguagens Irrestritas: Tipo 0.

- Implicação: existem mais problemas não-solucionáveis (infinitos, não-contáveis) do que problemas solucionáveis (infinitos, contáveis).
- Problema contável: que pode ser enumerado em uma sequência de números Naturais.
- Problemas contáveis:
  - $L = \{w\#w | w \in \{0,1\}^*\}$
  - Sucessor de um número em $\mathbb{N}$, $f(x) = x^2$, sequência de Fibonnaci, fatorial(x), etc.

Linguagens Recursivamente Enumeráveis

- Linguagens Recursivamente Enumeráveis são aquelas que podem ser aceitas por uma MT.
- É possível representar essas linguagens com um formalismo axiomático (gerador), no formato de uma Gramática.
- Essa Gramática é denominada *Gramática Irrestrita*.
    - A Gramática Irrestrita não possui qualquer restrição na forma das produções.

Linguagens Recursivamente Enumeráveis

- Uma linguagem aceita por uma MT é uma Linguagem Recursivamente Enumerável (LRE).
- A classe das LRE representa <u>todas</u> as linguagens que podem ser reconhecidas mecanicamente.
- Exemplos:
  - $L_1 = \{a^n b^n \mid n \geq 0\}$
  - $L_2 = \{w \mid w \text{ tem o mesmo número de símbolos } a \text{ e } b\}$
  - $L_3 = \{a^i b^j c^k \mid i = j \text{ ou } j = k\}$

Linguagens Recursivamente Enumeráveis

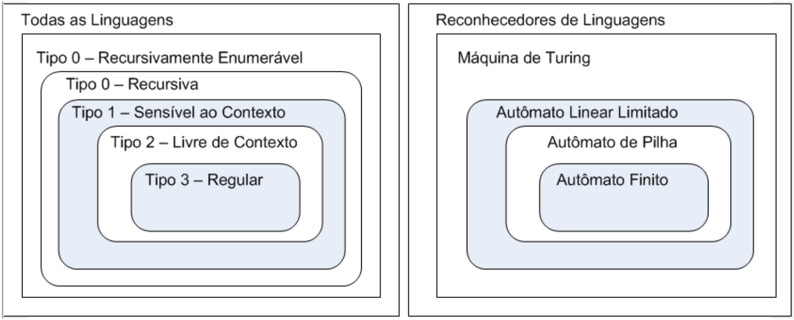

Figura: Classificação de Linguagens.

Linguagens Recursivamente Enumeráveis

- Distinguir uma MT em loop ou que está demorando para terminar é difícil.
- Por esse motivo, é preferível a MT que pára para todas as entradas.
- MT decisoras sempre tomam a decisão de aceitar ou rejeitar a entrada.
  - Toda linguagem Turing-decidível é Turing-reconhecível.
  - Nem toda linguagem Turing-reconhecível é Turing-decidível.
    - A MT Turing-reconhecível pode não parar para uma entrada não-aceita pela linguagem.

Linguagens Recursivamente Enumeráveis

- Exemplo: MT que decide a linguagem:
  $B = \{w\#w \mid w \in \{0\}^*\}$

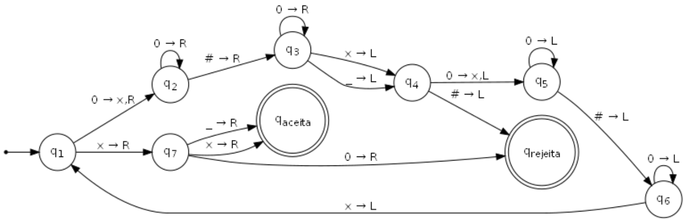

Figura: Diagrama de Estados da MT B.

Linguagens Recursivamente Enumeráveis

- Computação da palavra $w = 00\#00$:

$q_1 00\#00$	$x0\#0q_4 0\sqcup$	$xxq_2\#0x\sqcup$
$xq_2 0\#00$	$x0\#q_5 0x\sqcup$	$xx\#q_3 0x\sqcup$
$x0q_2\#00$	$x0q_5\#0x\sqcup$	$xx\#0q_3 x\sqcup$
$x0\#q_3 00$	$xq_6 0\#0x\sqcup$	$xx\#q_4 0x\sqcup$
$x0\#0q_3 0$	$q_6 x0\#0x\sqcup$	$xxq_5\#xx\sqcup$
$x0\#00q_3\sqcup$	$xq_1 0\#0x\sqcup$	$xq_6 x\#xx\sqcup$
		$xxq_1\#xx\sqcup$
		$xx\#q_7 xx\sqcup$
		$xx\#xq_{aceita}x\sqcup$

Seção 2

Linguagens Sensíveis ao Contexto

Linguagens Sensíveis ao Contexto

Linguagem Sensível ao Contexto (LSC)

Def.: Uma linguagem é:

- Tipo 1,
- Sensível ao Contexto,

se pode ser aceita por uma Máquina de Turing com fita limitada.

- O tamanho da fita é igual ao tamanho da entrada acrescido de duas células de controle.

Linguagens Sensíveis ao Contexto

- Formalismo axiomático: para gerar uma LSC é usada uma Gramática Sensível ao Contexto (GSC).
- O lado esquerdo das produções pode ser um símbolo não-terminal ou terminal, definindo um "contexto" da derivação.
    - Ex: $Aa \rightarrow B$ (onde a derivação de A depende do contexto a).
- A classe da LSC pertence às linguagens Turing-decidíveis.
- Inclui a grande maioria das linguagens aplicadas.

Linguagens Sensíveis ao Contexto

- Seja a Gramática Sensível ao Contexto (GSC):

$$G = (\Sigma, V_N, \mathbb{P}, Z)$$

- Σ: conjunto de símbolos terminais.
- V_N: conjunto de símbolos não-terminais.
- $\mathbb{P}$: conjunto de regras de produção (função parcial):

$$\alpha \rightarrow \beta$$

  - $\alpha \in (\Sigma \cup V_N)^+ \mid \ |\alpha| \leq |\beta|$, exceto para $Z \rightarrow \varepsilon$
- Portanto, em uma GSC, a cada derivação o tamanho da palavra <u>não</u> pode diminuir, exceto pela derivação que produz a palavra vazia.