INSTITUTO FEDERAL Ceará	Instituto Federal de Educação, Ciência e Tecnologia do Ceará
	Bacharelado em Engenharia de Computação
	Componente curricular: Aspectos Teóricos da Computação
	Docente: Francisco Nivando Bezerra
	Discente: Paulo Henrique de Sousa Melo

LISTA DE EXERCÍCIOS - REVISÃO DO CONTEÚDO

01. Como definimos linguagens regulares?

Uma linguagem é chamada de uma linguagem regular se algum autômato finito a reconhece.

02. O que são operações regulares? E expressões regulares?

Operações regulares são aquelas projetadas para manipular linguagens e estudar as suas propriedades. As operações regulares de união, concatenação e estrela (ou fecho) podem ser definidas conforme a imagem a seguir.

DEFINIÇÃO 1.23

Sejam A e B linguagens. Definimos as operações regulares ***união***, ***concatenação***, e ***estrela*** da seguinte forma.

- **União**: $A \cup B = \{x |\ x \in A \text{ ou } x \in B\}$.
- **Concatenação**: $A \circ B = \{xy |\ x \in A \text{ e } y \in B\}$.
- **Estrela**: $A^* = \{x_1 x_2 \ldots x_k |\ k \geq 0 \text{ e cada } x_i \in A\}$.

As expressões regulares, por sua vez, são aquelas que usam operações regulares para construir expressões descrevendo linguagens. Exemplo: (O∪1)0*. Essa expressão regular corresponde a uma linguagem que consiste de todas as cadeias começando com 0 ou 1 seguido por um número qualquer de 0s.

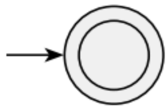
3. Que relação existe entre autômatos finitos e expressões regulares?

Expressões regulares e autômatos finitos são equivalentes no seu poder descritivo. Isso, porque qualquer expressão regular pode ser convertida em um autômato finito que reconhece a linguagem que ela descreve, e vice-versa.

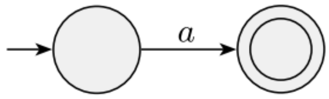
4. O que é a construção de Thompson? Como podemos transformar expressões regulares em autômatos equivalentes?

Trata-se de um algoritmo capaz de converter expressões regulares em autômatos equivalentes. Para realizar esse processo, segue-se o algoritmo de acordo com a operação existente na expressão regular.

Caso 1: R = ε

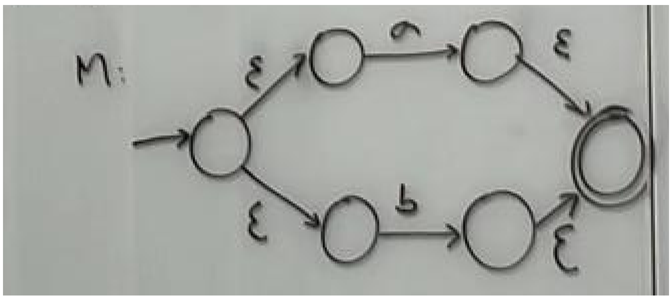


Caso 2: R = a

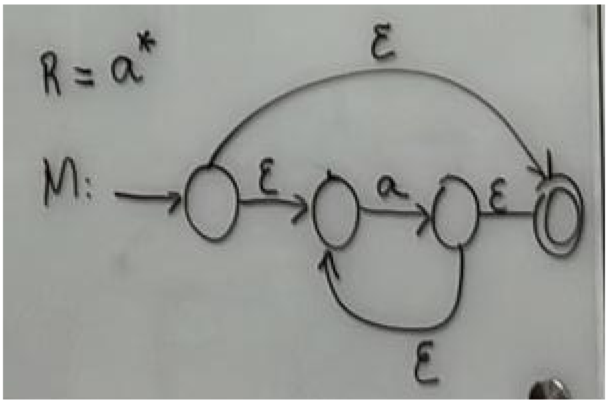


Caso 3: R = ab

Caso 4: R = a∪b



Caso 5: R = a*



5. Como definimos linguagens livres de contexto?

Antes de entender linguagens livres de contexto, é preciso compreender as gramáticas livres de contexto. Tais gramáticas são um método mais poderoso de escrever linguagens em comparação às gramáticas regulares. Elas podem descrever certas características que têm uma estrutura recursiva, o que as torna úteis em uma variedade de aplicações. Sabendo disso, chamamos de linguagens livres de contexto a coleção de linguagens associadas a gramáticas livres de contexto. Dessa forma, qualquer linguagem que pode ser gerada por alguma gramática livre de contexto é uma linguagem livre de contexto.

6. Qual a relação entre gramáticas livres de contexto e autômatos com pilha?

Os dois são equivalentes em poder, pois ambos são capazes de descrever a classe das linguagens livres de contexto. Assim, uma linguagem é livre de contexto se, e somente se, algum autômato com pilha a reconhece.

7. Dê exemplo de uma linguagem reconhecida por um autômato com pilha, mas não por uma expressão regular.

L = $\{0^n1^n \mid n \geq 0\}$. Um autômato finito é incapaz de reconhecer essa linguagem, porque ele não pode armazenar números muito grandes em sua memória finita. Um autômato com pilha tem a capacidade de reconhecer essa linguagem porque ele pode usar sua pilha para armazenar o número de 0s que ele já viu, isto é, a pilha é usada como uma memória. Portanto, a natureza ilimitada de uma pilha permite ao autômato com pilha armazenar números de tamanho ilimitado.

8. O que é derivação mais à esquerda? E gramática ambígua?

Derivação mais à esquerda é o processo em que, a cada passo da derivação, a variável mais à esquerda remanescente é aquela a ser substituída. Quando uma cadeia tem duas ou mais derivações mais à esquerda, diz-se que ela foi derivada ambiguamente. A gramática associada a essa linguagem é, portanto, ambígua, pois ela gerou uma cadeia ambiguamente.

9. Dê exemplo de linguagem que pode ser gerada por uma gramática livre e contexto, mas não por uma expressão regular.

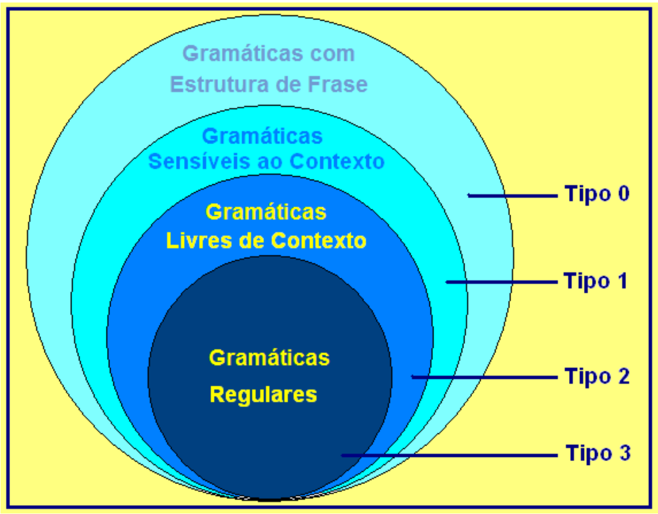
L = $\{0^n 1^n \mid n \geq 0\}$. O mesmo raciocínio da questão 7 pode ser aplicado no cenário da questão 9, além de que essa linguagem não representa uma expressão regular.

10. Qual a relação entre linguagens regulares e linguagens livres de contexto?

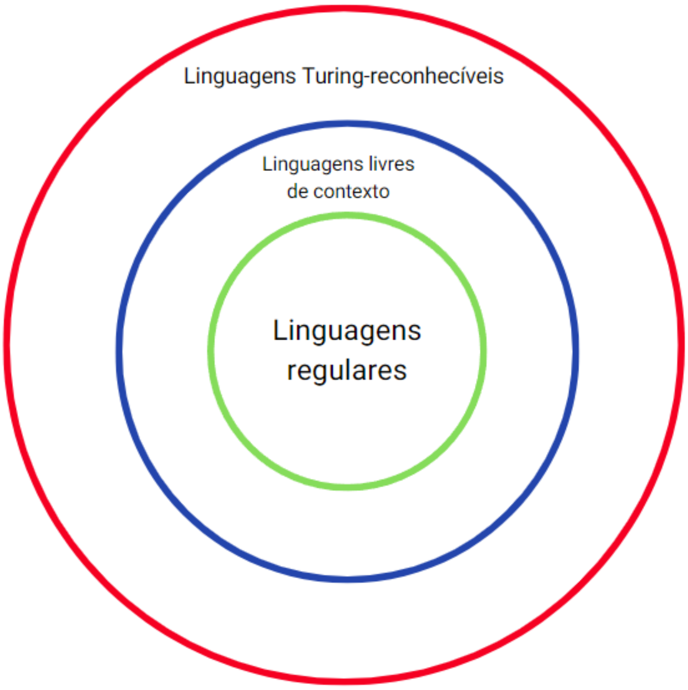
As linguagens livres de contexto estão em um nível superior da hierarquia de Chomsky em relação às linguagens regulares, por serem mais poderosas e poderem representar estruturas que não podem ser representadas por expressões regulares.

11. O que é hierarquia de Chomsky?

Trata-se de um modo de organizar hierarquicamente linguagens de acordo com o seu poder. A imagem a seguir representa uma hierarquia de Chomsky.



No contexto da disciplina de Aspectos Teóricos da Computação, essa hierarquia pode ser resumida do seguinte modo, seguindo a estrutura montada pelo professor em sala:



12. O que são linguagens Turing-reconhecíveis? E linguagens Turing-decidíveis?

Uma linguagem é dita Turing-reconhecível se alguma máquina de Turing a reconhece. Por outro lado, uma linguagem é chamada de Turing-decidível se alguma máquina de Turing a decide. Essa máquina é nomeada decisor e nunca entra em loop, isto é, ela sempre decide entre aceitação e rejeição.

13. Qual a relação entre algoritmos e máquinas de Turing? O que é a tese de Church-Turing?

A máquina de Turing serve meramente como um modelo preciso para a definição de algoritmo. A tese de Church-Turing defende que "toda função que seria naturalmente considerada computável pode ser computada por uma máquina de Turing".

14. Como você pode enunciar o 10º problema de Hilbert?

O problema consiste em projetar um algoritmo que testa se um polinômio tem uma raiz inteira. Ele não é Turing-decidível, mas é Turing-reconhecível.

15. Enuncie o problema da parada.

O problema da parada consiste em elaborar um algoritmo que seja capaz de inspecionar o código de outro algoritmo e determinar se tal algoritmo entra em loop ou não quando lhe é dada determinada entrada. Trata-se de um problema indecidível.

REFERÊNCIAS

SIPSER, Michael. **Introdução à teoria da computação**. São Paulo (SP): Cengage Learning, 2011. 459 p. Tradução da 2ª edição americana.