



UniRuy & Área 1 | Wyden  
PROGRAMA DE CIÊNCIA DA COMPUTAÇÃO  
TEORIA DE COMPILADORES

JOÃO MARCELO TAVARES SOUZA MATOS

Linguagens, Expressões e Gramáticas Regulares

Salvador - Bahia - Brasil

2022

JOÃO MARCELO TAVARES SOUZA MATOS

## Linguagens, Expressões e Gramáticas Regulares

Trabalho Acadêmico elaborado junto ao programa de Engenharia UniRuy & Área 1 | Wyden, como requisito para obtenção de nota parcial da AV1 na disciplina Teoria de Compiladores no curso de Graduação em Ciência da Computação, que tem como objetivo consolidar os tópicos do plano de ensino da disciplina.

Orientador: Prof. MSc. Heleno Cardoso

Salvador - Bahia - Brasil

2022

da Tal, Aluno Fulano

Teoria de Compiladores: Resenha / Mapa Mental / Perguntas

– Aluno Fulano de Tal. Salvador, 2022.  
18 f. : il.

Trabalho Acadêmico apresentado ao Curso de Ciência da Computação, UniRuy & Área 1 | Wyden, como requisito para obtenção de aprovação na disciplina Teoria de Compiladores.

Prof. MSc. Heleno Cardoso da S. Filho.

1. Resenha
2. Mapa Mental
3. Perguntas/Respostas (Mínimo de 03 – Máximo de 05)
4. Conclusão

I. da Silva Filho, Heleno Cardoso II. UniRuy & Área 1  
| Wyden. III. Trabalho Acadêmico

CDD:XXX

# TERMO DE APROVAÇÃO

JOÃO MARCELO TAVARES SOUZA MATOS

LINGUAGENS, EXPRESSÕES E GRAMÁTICAS REGULARES

Trabalho Acadêmico aprovado como requisito para obtenção de nota parcial da AV2 na disciplina Teoria de Compiladores, UniRuy & Área 1 | Wyden, pela seguinte banca examinadora:

BANCA EXAMINADORA

Prof<sup>o</sup>. MSc<sup>o</sup>. Heleno Cardoso  
Wyden

Salvador, 09 de Outubro de 2022

Dedico este trabalho acadêmico a todos que contribuíram direta ou indiretamente com  
minha formação acadêmica.

## **Agradecimentos**

Primeiramente agradeço a Deus. Ele, sabe de todas as coisas, e através da sua infinita misericórdia, se fez presente em todos os momentos dessa trajetória, concedendo-me forças e saúde para continuar perseverante na minha caminhada.

E a todos aqueles que contribuíram direta ou indiretamente para a minha formação acadêmica.

*"A educação tem raízes amargas, mas os seus frutos são doces".*

*Aristóteles.*

## Resumo

Graças às teorias da computação, sobretudo as teorias dos autômatos, onde é também chamada de Máquina de estado finita, gerou para a computação varios beneficios, como analise lexicas, reconhecimento de padrões e ajudou a desenvolver tanto hardwares e tanto os softwares.

Palavras-chaves: Teoria da computação, autômatos, analise lexica, hardware, software



## **Abstract**

Thanks to the theories of computation, especially the theories of automata, where it is also called a finite state machine, it generated several benefits for computing, such as lexical analysis, pattern recognition and helped to develop both hardware and software.

Keywords: Theory of computation, automata, lexical analysis, hardware, software

## Lista de figuras

Figura 1 – Operações Regulares . . . . .	15
Figura 2 – Operações Regulares - União . . . . .	15
Figura 3 – Operações Regulares - União - Exemplo . . . . .	16
Figura 4 – Operações Regulares - Concatenação . . . . .	16
Figura 5 – Operações Regulares - Concatenação - Exemplo . . . . .	16
Figura 6 – Operações Regulares - Kleene-+ . . . . .	17
Figura 7 – Operações Regulares - Kleene-+ - Exemplo . . . . .	17
Figura 8 – Operações Regulares - Kleene-* . . . . .	18
Figura 9 – Operações Regulares - Kleene-* - Exemplo . . . . .	18

## Lista de abreviaturas e siglas

AFNs - Automatos Finitos não determinísticos

ER - Expressões regulares

AFD - Automatos finitos determinísticos

# Sumário

<b>1</b>	<b>Linguagens, Expressões e Gramáticas Regulares</b>	<b>12</b>
1.1	Introdução	12
1.1.1	Perguntas e Respostas - Mínimo de 2 e Máximo de 5	12
1.2	Conclusão	12
<b>2</b>	<b>Definição de linguagem regular</b>	<b>13</b>
2.1	Definição	13
2.2	Identificando linguagens não regulares	13
2.3	Lema do Bombeamento para Linguagens Regulares	13
<b>3</b>	<b>Definição de expressão regular</b>	<b>14</b>
<b>4</b>	<b>Operações regulares</b>	<b>15</b>
4.1	União	15
4.2	Concatenação	16
4.3	Kleene-+	17
4.4	Kleene-*	17
<b>5</b>	<b>Definição de gramática regular</b>	<b>19</b>
<b>6</b>	<b>Equivalência entre gramática regular</b>	<b>20</b>
<b>7</b>	<b>Autômato finito e expressão regular</b>	<b>21</b>
	<b>Referências<sup>1</sup></b>	<b>22</b>

---

<sup>1</sup> De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

# 1 Linguagens, Expressões e Gramáticas Regulares

## 1.1 Introdução

Diante da evolução da ciência da computação, as linguagens, os softwares e hardwares foram desenvolvidos a partir de diversas teorias e teoremas, alguns deles, foram desenvolvidos a partir das linguagens e expressões regulares.

Tanto a linguagem como as expressões estão diretamente ligadas a análise sintática de um compilador, assim como os AFD e AFN(Autômato Finitos determinísticos e não determinísticos) estão ligados à análise Lexica que também é uma das fases de compilação.

### 1.1.1 Perguntas e Respostas - Mínimo de 2 e Máximo de 5

1° Pergunta: Qual análise é ligada aos Automatos finitos?

2° Pergunta: No conceito de programação, as expressões regulares, hoje como utilidade dentro de um código orientado a objetos, servem para qual finalidade?

## 1.2 Conclusão

Conforme os tópicos citados nesse artigo, tanto as linguagens, como as expressões regulares e automatos finitos tem ligação direta com o desenvolvimento das linguagens compiladas, sendo eles responsáveis pela análise do código, sem o desenvolvimento dessas teorias, hoje não poderíamos ter diversas evoluções científicas feitas a partir da criação dessas teorias.

## 2 Definição de linguagem regular

### 2.1 Definição

De acordo com Newton José Viera em seu artigo: A Linguagem regular é dita ser uma linguagem regular se existe um autômato finito que a reconhece.

Dada uma linguagem  $L$ :

- É possível determinar se ela pertence ou não à classe das linguagens regulares?
- É possível facilitar a obtenção de um AF para  $L$ ? ([VIEIRA](#), )

### 2.2 Identificando linguagens não regulares

As linguagens regulares podem ser infinitas. No entanto, o fato delas poderem ser associados com autômatos que tem memória finita, impõe alguns limites na estrutura das linguagens regulares. Nossa intuição diz que uma linguagem é regular somente se, em processando qualquer cadeia, a informação a ser armazenada em qualquer estágio é estritamente limitada. Isso é verdade, mas veremos de mostrar precisamente. ([BEDEGRAL](#), 2008)

### 2.3 Lema do Bombeamento para Linguagens Regulares

O resultado que segue, conhecido como lema do bombeamento para linguagens regulares, usa o princípio da casa de pombos numa outra forma. A prova é baseada na observação de que num grafo de transição com  $n$  vértices, qualquer caminho de comprimento  $n$  ou mais longo deve repetir algum vértice, isto é, contém um ciclo. O lema do bombeamento só precisará analisar linguagens infinitas, pois trivialmente toda linguagem finita necessariamente é regular. ([BEDEGRAL](#), 2008)

### 3 Definição de expressão regular

Uma expressão regular é uma notação para representar padrões em strings. Serve para validar entradas de dados ou fazer busca e extração de informações em textos. Por exemplo, para verificar se um dado fornecido é um número de 0,00 a 9,99 pode-se usar a expressão regular `\d,\d, \d`, pois o símbolo `\d` é um curinga que casa com um dígito. O verbo casar aqui está sendo usado tradução para match, no sentido de combinar, encaixar, parear. Dizemos que a expressão `\d,\d, \d` casa com 1,2 3 mas não casa com 123 (falta a vírgula) nem com 1,2c (“c” não casa com `\d`, porque não é um dígito). O termo em inglês é regular expression de onde vem as abreviações regex. Na ciência da computação, o termo tem um significado bem específico. ([RAMALHO, 2012](#))

Exemplos:

`[A-Z] {3}\d{4}`

É o padrão de uma placa de automóvel no Brasil: três letras de A a Z é seguidas de um - (hífen) seguido de quatro dígitos, como CKD-4592. ([RAMALHO, 2012](#))

## 4 Operações regulares

O círculo vermelho representa o estado inicial  $q_0$ , a porção verde representa o conjunto de estados de aceitação  $F$ , os demais estados são a região cinza (ANDERSON, 2011)

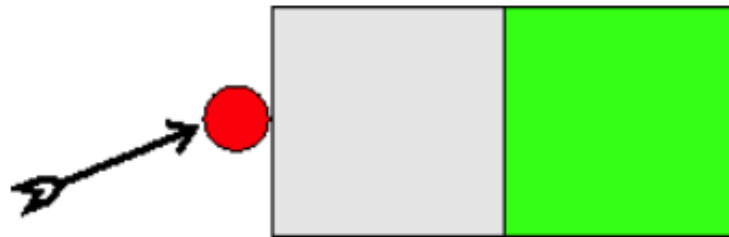


Figura 1 – Operações Regulares

### 4.1 União

A união  $A \cup B$  é formada colocando os autômatos em paralelo. Criamos um novo estado inicial e o conectamos aos estados iniciais originais por transição: (ANDERSON, 2011)

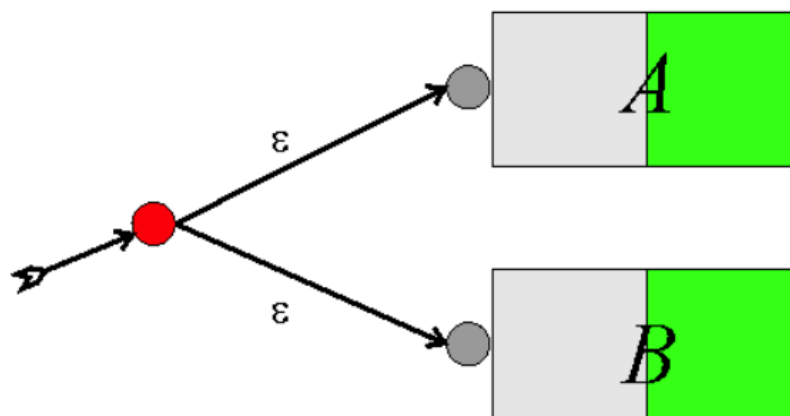


Figura 2 – Operações Regulares - União

Exemplo:

$$L = \{ x \text{ tem comprimento par} \} \cup \{ x \text{ termina com } 11 \}$$



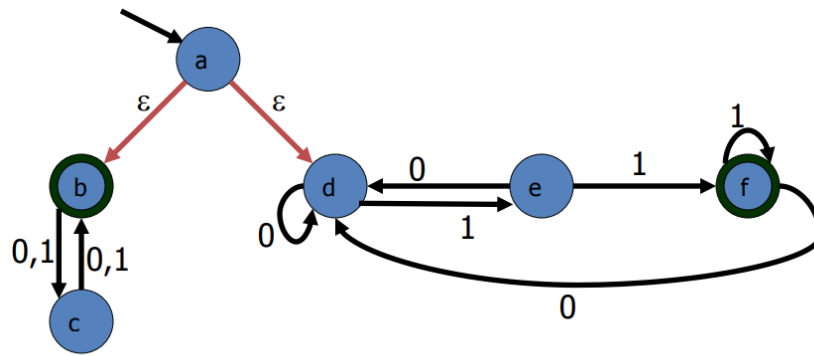


Figura 3 – Operações Regulares - União - Exemplo

## 4.2 Concatenação

A concatenação  $A \bullet B$  é formada colocando-se os autômatos em série. O estado inicial é o de A enquanto os estados de aceitação são os de B. Os estados de aceitação de A deixam de sê-lo e são conectados, por meio  $\epsilon$ -transição, ao estado inicial de B: (ANDERSON, 2011)

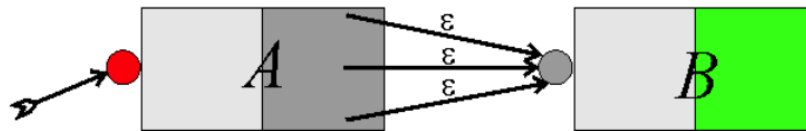


Figura 4 – Operações Regulares - Concatenação

Exemplo:

$$L = \{ x \text{ tem comprimento par} \} \bullet \{ x \text{ termina com } 11 \}$$

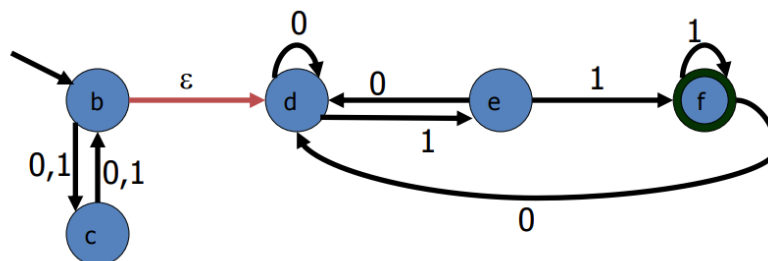


Figura 5 – Operações Regulares - Concatenação - Exemplo

### 4.3 Kleene-+

A operação Kleene-+  $A^+$  é formada criando-se um loop de retroalimentação. Os estados de aceitação são conectados ao estado inicial por meio de e-transição: (ANDERSON, 2011)

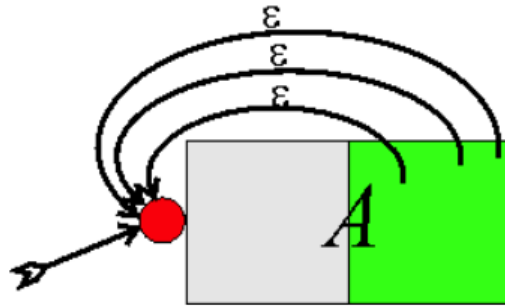


Figura 6 – Operações Regulares - Kleene-+

Exemplo:

$L = \{ x \text{ é uma sequência de 1 ou mais 1's seguida de uma sequência de 2 ou mais 0's} \}^+$   
 $= \{ x \text{ começa com 1, termina com 0, e consiste de repetições de sequências de 1 ou mais 1's seguidos de 2 ou mais 0's} \}$

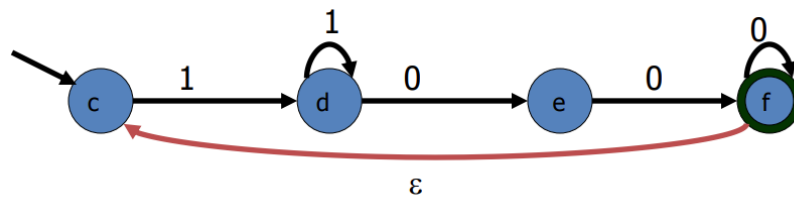


Figura 7 – Operações Regulares - Kleene-+ - Exemplo

### 4.4 Kleene-\*

A construção deriva da de Kleene-+ usando-se o fato de que  $A^*$  é a união de  $A^+$  com a linguagem que contém apenas o string vazio. Construa Kleene-+ e adicione um novo inicial conectado ao antigo estado inicial por meio de e-transição: (ANDERSON, 2011)

Exemplo:

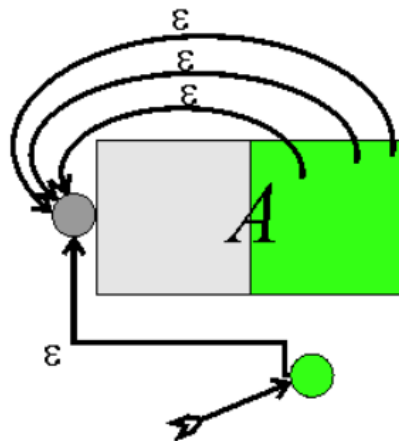


Figura 8 – Operações Regulares - Kleene-\*

$L = \{ x \text{ é uma sequência de 1 ou mais 1's seguida de uma sequência de 2 ou mais 0's} \}^*$

$= \{ x \text{ começa com 1, termina com 0, e consiste de repetições de sequências de 1 ou mais 1's seguidos de 2 ou mais 0's ou } x \text{ é } \epsilon \}$

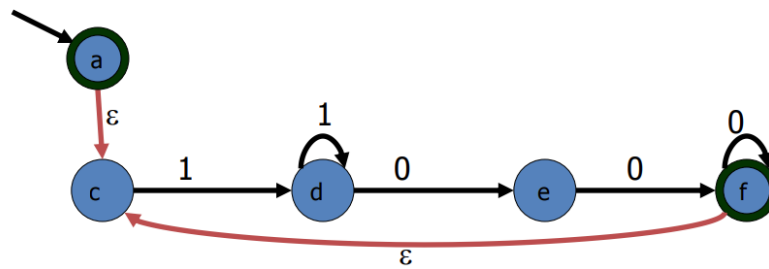


Figura 9 – Operações Regulares - Kleene-\* - Exemplo

## 5 Definição de gramática regular

Em Teoria da computação as Gramáticas regulares também conhecida como Tipo 3 da Hierarquia de Chomsky, é uma restrição sobre a forma das produções, pode-se criar uma nova classe de gramáticas de grande importância no estudo dos compiladores por possuírem propriedades adequadas para a obtenção de reconhecedores simples. Que também podem ser denominada de Expressão regular. (RUIZ, 2017)

São gramáticas mais restritas e mais próximas das formas de produção correspondentes às transições de autômatos finitos.

Definição: Uma gramática  $G = (VN, VT, S, P)$  diz-se regular (GR), ou do tipo 3, se as suas produções são da forma:  $N \rightarrow tM$  ou  $N \rightarrow t'$ , onde  $N, M \in VN$  e  $t, t' \in VT$  e  $t \neq \epsilon$ . (TAVARES, 2000)

## 6 Equivalência entre gramática regular

Definição (Equivalência de gramáticas): Duas gramáticas são equivalentes se, e somente se,  $L(G') = L(G'')$ . ([TAVARES, 2000](#))

## 7 Autômato finito e expressão regular

AFs e Expressões regulares são equivalentes em seu poder descritivo. Qualquer expressão regular pode ser convertida num autômato finito que reconhece a linguagem que ela descreve, e vice versa.

Teorema: Uma linguagem é regular se e somente se alguma expressão regular a descreve. Esse teorema tem duas direções. Enunciamos e provamos cada uma das direções como um lema separado.

- Se uma linguagem é descrita por uma expressão regular então ela é regular.
- Se uma linguagem é regular então ela é descrita por uma expressão regular.

(CAMBUIM, S.A)

## Referências<sup>1</sup>

- ANDERSON. *Operações Regulares e Expressões Regulares*. 2011. Citado 3 vezes nas páginas 15, 16 e 17.
- BEDEGRAL. *Propriedades das Linguagens Regulares*. 2008. Citado na página 13.
- CAMBUIM, L. *EQUIVALÊNCIA COM AUTÔMATOS FINITOS*. S.A. Citado na página 21.
- RAMALHO, L. *Expressões regulares: introdução*. 2012. Citado na página 14.
- RUIZ, E. E. S. *Gramática regular*. 2017. Citado na página 19.
- TAVARES. *Expressões Regulares e Gramáticas*. 2000. Citado 2 vezes nas páginas 19 e 20.
- VIEIRA, N. J. *Linguagens Regulares: Conclusão*. Citado na página 13.

---

<sup>1</sup> De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.