

Licenciatura em Engenharia Informática – DEI/ISEP  
**Linguagens e Programação 2020/2021**  
Aula Prática-Laboratorial

# Ficha PL 1

## Expressões Regulares e Autómatos Finitos

### Objetivos:

- Introdução ao reconhecimento de padrões e expressões regulares;
- Exercícios com expressões regulares com vista à consolidação dos conhecimentos introduzidos na Ficha TP 1;
- Classificação de Autómatos Finitos (AF): Determinísticos (AFD) ou Não Determinísticos (AFN);
- Validação de palavras utilizando Autómatos Finitos;
- Conversão de autómatos finitos não determinísticos (AFN) em autómatos finitos determinísticos (AFD);
- Minimização de Autómatos Finitos Determinísticos;
- Consolidação dos conceitos apresentados na Ficha TP2, através da realização de exercícios.

### 1. EXPRESSÕES REGULARES

As expressões regulares (ER) constituem um método formal para especificar um padrão de texto. São regras bem definidas que permitem gerar um conjunto de sequências de símbolos (palavras). Traduzem um conjunto de padrões, eventualmente de difícil e complexa enumeração, numa expressão de dimensão curta e relativamente fácil de interpretar. Uma ER constrói-se com base num conjunto de meta-caracteres (\*, [], ^, ., -, ?, entre outros) que são interpretados de forma especial, escritos numa dada sequência e/ou intercalados com sequências de caracteres sem significado especial que não seja a sua verificação.

1. Descreva informalmente (por palavras) as linguagens representadas pelas seguintes expressões regulares:
  - a)  $0(0|1)^*0$
  - b)  $(01)^*$
  - c)  $(0|1)^*0(0|1)(0|1)$
  - d)  $0^*10^*10^*10^*$
  - e)  $(0|1)^*111(0|1)^*$
2. Considerando o alfabeto  $\Sigma = \{A, B, C, \dots, Z, a, b, c, \dots, z\}$ , represente as seguintes linguagens utilizando expressões regulares:
  - a)  $L(A) = \{u \in \Sigma^* : u \text{ tem pelo menos dois } a\text{'s juntos}\}$
  - b)  $L(A) = \{u \in \Sigma^* : u \text{ tem comprimento inferior a } 4\}$

- c)  $L(A) = \{u \in \Sigma^* : u \text{ começa por uma maiúscula}\}$   
d)  $L(A) = \{u \in \Sigma^* : u \text{ tem um número par de maiúsculas e um número ímpar de minúsculas, e tem comprimento múltiplo de 4}\}$
3. Diga se são verdadeiras ou falsas as seguintes afirmações:
- a)  $(0|1)^* \Leftrightarrow (0^*|1^*)$   
b)  $(0|1)^* \Leftrightarrow (0^*1^*)^*$
4. Considerando o alfabeto  $\Sigma = \{0,1\}$ , represente as seguintes linguagens utilizando expressões regulares:
- a)  $L(A) = \{u \in \Sigma^* : u \text{ é um número binário múltiplo de 4}\}$   
b)  $L(A) = \{u \in \Sigma^* : 111 \text{ não é factor de } u\}$   
c)  $L(A) = \{u \in \Sigma^* : u \text{ tem um número par de 1's}\}$   
d)  $L(A) = \{u \in \Sigma^* : u \text{ é vazia ou tem dígitos todos iguais, sendo de comprimento par as sequências de 0's, e ímpar as sequências de 1's}\}$   
e)  $L(A) = \{u \in \Sigma^* : u \text{ é múltiplo de 4 mas não de 8}\}$   
f)  $L(A) = \{u \in \Sigma^* : u \text{ tem comprimento 3}\}$
5. Para a classificação dos alunos de uma escola, cada nota encontra-se codificada através do par NT, em que N é um valor inteiro entre 0 e 20 e T pode ser a letra F, ou E ou o par de letras RF. Uma nota não pode conter um zero na casa das dezenas. Uma sequência de notas, terá de conter pelo menos uma nota. Escrever uma expressão regular que valide uma sequência ilimitada de notas.
6. Para garantir a rastreabilidade dos produtos fabricados numa determinada unidade industrial, cada lote de fabrico deverá ser codificado de forma a representar a informação necessária. Os campos que devem ser definidos nesta codificação são os definidos na tabela 1.1.

**Tabela 1.1:** Representação da codificação dos lotes de fabrico

Campo	Tamanho	Descrição
Identificação do lote	5	2 dígitos para o ano seguidos de 3 para o ID antecidos dos caracteres LT
Referência da peça	4	dígitos alfanuméricos
Identificação da máquina	2	2 dígitos
Quantidade	5	5 dígitos
Data (Mês/Dia)	4	2 dígitos para o mês (01-12) seguidos de 2 para o dia (01-31)

Escreva uma Expressão Regular que permita representar a codificação dos lotes de fabrico. A ordem a seguir deve ser a definida na tabela anterior e deve ser usado o hífen como separador. Ver exemplo seguinte:

**LT16121-3z1a-23-13450-1205**

7. Considerando A empresa DEI-TB precisa de apoio técnico para operacionalizar a informação apresentada no seu canal televisivo, durante os jogos de futebol do EURO2016. O sistema é suportado por um ficheiro TXT com informação dos vários jogos a transmitir. Entre outras, a informação comporta a estrutura **PT~MM:SS~EQP>G-G<EQP**, onde:

- PT, é opcional, e corresponde a uma das palavras 1P, 2P, 1EP ou 2EP
- MM corresponde à componente a minutos, podendo ter com 2 ou 3 dígitos (valores entre 00 e 120)
- SS corresponde a à componente segundos, com 2 dígitos (entre 00 e 59)
- EQP corresponde à sigla duma equipa, composta por 3 letras maiúsculas
- G corresponde aos golos marcados duma por uma equipa com valor inferior a 10

NOTA: o tempo máximo de jogo é 120 minutos!

**Exemplos:**

... 89:12~FRA>2-1<ROM ... 2P~28:06~ENG>1-0<RUS ...  
... 1P~03:45~ESP>0-0<ROM ... EP~101:37~POR>4-2<ISL ...

Construa uma expressão regular para captar a informação desejada.

## 2. Autómatos Finitos

Um autómato finito ou máquina de estados é um formalismo que permite representar, de forma clara, um qualquer processo composto por um conjunto de estados e transições entre esses estados. Os autómatos finitos podem ser utilizados para descrever o processo de reconhecimento de padrões em cadeias de entrada.

- Para a linguagem constituída por todas as strings no alfabeto {a, b, c, d} em que o primeiro "b" (se existir) é precedido de um "a";
  - Represente graficamente um autómato capaz de reconhecer a linguagem.
  - Defina formalmente o autómato A de acordo com a fórmula  $A = (S, \Sigma, s_0, F, \sigma)$ .
  - Elabore a tabela de transições respetiva.
  - Represente essa linguagem usando uma expressão regular.
- Considere a seguinte expressão regular: **(a|b)\*abb**
  - Represente graficamente o autómato correspondente.
  - Defina formalmente o autómato A de acordo com a fórmula  $A = (S, \Sigma, s_0, F, \sigma)$ .
  - Elabore a tabela de transições respetiva.
- Represente um autómato finito capaz de reconhecer números reais no seguinte formato: 'inteiro','inteiro'. O número real a reconhecer deverá ter sempre uma vírgula e uma casa decimal, não sendo obrigatória a existência de uma parte inteira.

4. Analise o seguinte código escrito em linguagem C e compare-o com o autómato desenvolvido no exercício 3, retire conclusões:

```

1 #include<stdio.h>
2
3 int main()
4 {
5     int a;
6     int real=1;
7     char estado='A' ;
8     int erro=0;
9
10    printf("Introduza dados:");
11    while ((a=getchar())!='\n') && (erro==0))
12        switch (estado)
13        {
14            case 'A' :
15                if (( a>='0') && (a<='9'))
16                    estado='A';
17                else
18                    if ( a==',' )
19                        estado='B';
20                    else
21                        erro=1;
22            break;
23
24            case 'B' :
25                if((a>='0') && (a<='9'))
26                    estado='C';
27                else
28                    erro=1;
29            break ;
30
31            case 'C' :
32                if((a>='0') && (a<='9'))
33                    estado='C' ;
34                else
35                    erro =1;
36            break;
37        }
38    if ((!erro) && (estado=='C'))
39        printf ( "Foi lido um real.\n" ) ;
40    else
41        printf ("Não foi lido um real.\n" ) ;
42    return 0
43 }
```

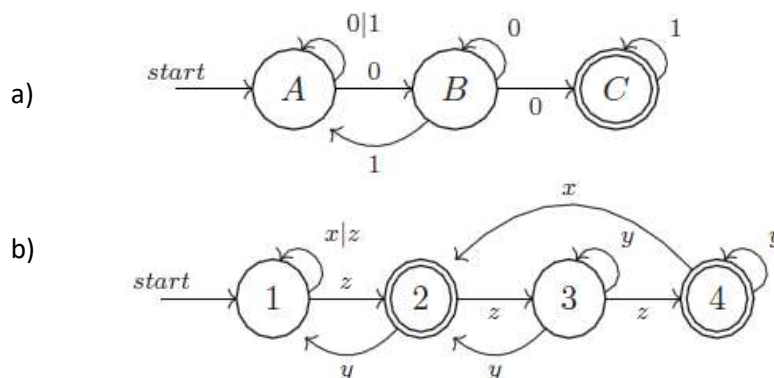
5. As quantias monetárias podem ser expressas de diferentes formas.

**Tabela 2.1:** Representação de moedas

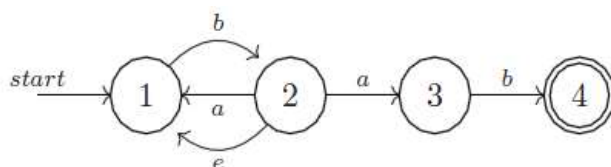
Moeda	Exemplo
Euro	e12,23; e1,00; e2,35; 23,50EUR
Libra	£12.50; £22.12; £22.99
Dólar	\$25.13; \$5.00; \$0.30
Escudo	12\$50; 25\$00; 150\$00; 0\$50

- Escreva as expressões regulares que permitam realizar o reconhecimento das moedas apresentadas na tabela 2.1.
- Crie um AF capaz de identificar se uma quantia monetária está de acordo com a tabela 2.1.

6. Considerando o alfabeto  $\Sigma = \{A, B, C, \dots, Z, a, b, c, \dots, z\}$ , represente as seguintes linguagens utilizando um AF:
- $L(A) = \{u \in \Sigma^* : u \text{ começa por um carácter minúsculo e tem no máximo duas maiúsculas}\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ começa por uma maiúscula e termina sempre numa vogal}\}$
7. Considerando o alfabeto  $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ , represente as seguintes linguagens utilizando um AF:
- $L(A) = \{u \in \Sigma^* : u \text{ é uma sequência numérica crescente}\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ é múltiplo de 5}\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ é um número ímpar}\}$
8. Considerando o alfabeto  $\Sigma = \{0, 1\}$ , represente as seguintes linguagens utilizando um AF. Classifique, justificando, os AF obtidos:
- $L(A) = \{u \in \Sigma^* : u \text{ é um número binário múltiplo de 4}\}$
  - $L(A) = \{u \in \Sigma^* : 111 \text{ é factor de } u\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ tem um número par de 1's}\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ é vazia ou tem dígitos todos iguais, sendo de comprimento par as sequências de 0's, e ímpar as sequências de 1's}\}$
9. Represente graficamente e através da tabela de transições, os AF capazes de reconhecer as seguintes expressões regulares e indique, **justificando**, se são determinísticos ou Não Determinísticos:
- $aa^*|bb^*$
  - $(a|b)^*abb(a|b)^*$
10. Converta os seguintes AFN num AFD e minimize-os.



11. Indique uma expressão regular e um autómato finito determinístico que reconheça a mesma linguagem que o seguinte autómato finito não determinístico.



12. Considere a seguinte tabela de transições:

	0	1
→S	A	F
A	C	B
B	F	A
C	A	F
*F	B	C

- Represente graficamente o autómato finito correspondente. Classifique o autómato resultante.
- Verifique se as seguintes palavras pertencem à linguagem reconhecida pelo autómato:
  - 1
  - 0110110
  - 001
  - 1001
  - 10101
  - 001101
- Minimize o autómato finito da alínea a).
- Descreva informalmente (por palavras) as linguagens representadas pelo autómato.

13. Implemente um autómato finito determinístico para cada uma das expressões regulares:

- $[0-9]^+$
- $[0-9]^+ \cdot "[0-9]^+$
- $[f-o][k-z3-7]^*$
- $("---"[a-z]^*\backslash n") | (" " | "\n" | "\t")$

### 3. Exercícios complementares

- Considerando o alfabeto  $\Sigma = \{0,1\}$ , represente as seguintes linguagens utilizando expressões regulares:
  - $L(A) = \{u \in \Sigma^* : 10 \text{ é prefixo de } u\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ é um número binário que ou não é par ou é múltiplo de } 8\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ tem no máximo comprimento } 3 \text{ e é uma capicua}\}$
- Considerando o alfabeto  $\Sigma = \{0,1,2,3,4,5,6,7,8,9\}$ , represente as seguintes linguagens utilizando expressões regulares:
  - $L(A) = \{u \in \Sigma^* : u \text{ é uma sequência numérica crescente}\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ começa e termina no mesmo algarismo}\}$
- Considere o extrato seguinte de um ficheiro que contém informação georreferenciada sobre um conjunto de cidades.

```
# Cidades
41°9'N      8°38'W      11-02-2007   9:10:34      Porto
38°42'N      9°11'W      11-02-2007   9:10:42      Lisboa
51°30'25"N   0°07'39"W   11-02-2007   9:10:43      Londres
22°54'30"S   43°11'47"W   11-02-2007   9:10:43      Rio
55°45'8"N    37°37'56"E   11-02-2007   9:10:45      Moscovo
#Trip description
```

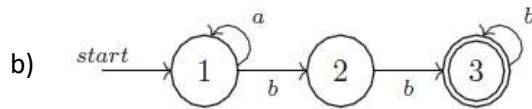
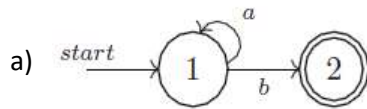
8. Escreva as expressões regulares que permitem a identificação dos campos apresentados na tabela 2.2. Considere também a necessidade de reconhecer as linhas de comentário (iniciadas com o carácter #).

**Tabela 2.2:** Campo de informação geo-referenciada

Campo	Exemplo
Latitude	41°9'N
	22°54'30"S
Longitude	9°11'W
	37°37'56"E
Data	11-02-2007
Hora	9:10:34
Cidade	Porto

9. Represente graficamente e através da tabela de transições, os AF capazes de reconhecer as seguintes expressões regulares e indique se são Determinísticos ou Não Determinísticos:
- $(a)(a^*|b^*)^*$
  - $(b)a(a|b)^*$
10. Considerando o alfabeto  $\Sigma = \{0, 1\}$ , represente as seguintes linguagens utilizando um AF:
- $L(A) = \{u \in \Sigma^* : u \text{ é um número binário que ou não é par ou é múltiplo de } 8\}$
  - $L(A) = \{u \in \Sigma^* : u \text{ é múltiplo de } 4 \text{ mas não de } 8\}$
11. Represente graficamente e através da tabela de transições, os AFN capazes de reconhecer a linguagens constituída pelas *strings* no alfabeto  $\{a, b, c, \dots, z\}$  que contêm as cinco vogais em sequência e por ordem alfabética pelo menos 1 vez.
- xaeiouz** - válido  
**xaaeiouz** - válido  
**xaeiouz** - inválido  
**xayeiouz** - inválido  
**xayaeiouwez** - válido  
**xaeiouyaeiouz** – válido
12. Considerando o alfabeto  $\Sigma = \{A,B,C, \dots, Z; a, b, c, \dots, z\}$ , represente a seguinte linguagem utilizando um AF:
- $$L(A) = \{u \in \Sigma^* : u \text{ tem pelo menos dois } a\text{'s juntos}\}$$
13. Considerando o alfabeto  $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ , represente a seguinte linguagem utilizando um AF:
- $$L(A) = \{u \in \Sigma^* : u \text{ é potência de } 10\}$$
14. Implemente um autómato finito determinístico para cada uma das seguintes linguagens:
- Para o alfabeto  $\Sigma = \{a, b, c, d\}$ , qualquer palavra com um número par de símbolos "b".
  - Para o alfabeto  $\Sigma = \{0, 1\}$ , qualquer palavra com um número par de zeros e um número par de uns.

15. Para cada um dos seguintes autómatos finitos determinísticos indique uma expressão regular que reconheça a mesma linguagem.



16. Represente as seguintes linguagens por uma expressão regular e por um AFD:

- a) Para o alfabeto  $\Sigma = \{0, 1\}$

$L(A) = \{u \in \Sigma^* : u \text{ começa por } 1 \text{ e tem um número par de } 1\text{'s}\}$

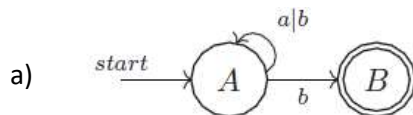
- b) Para o alfabeto  $\Sigma = \{+, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, e, i, .\}$

$L(A) = \{u \in \Sigma^* : u \text{ é um número imaginário}\}$

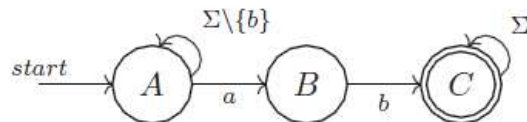
- c) Para o alfabeto  $\Sigma = \{I, V, X, L, C\}$

$L(A) = \{u \in \Sigma^* : u \text{ é um número romano menor que } 100\}$

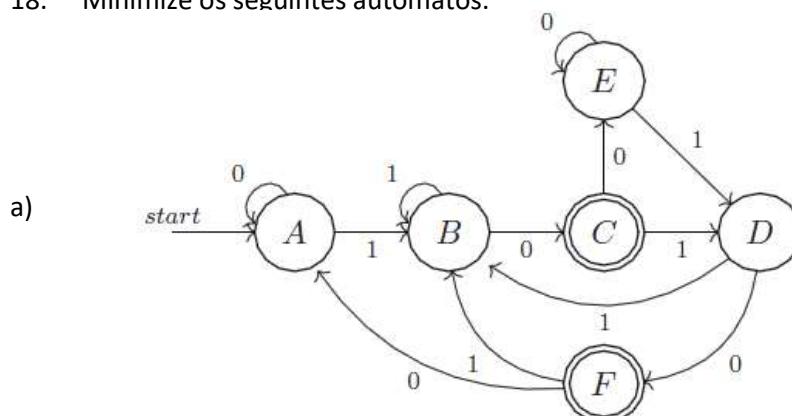
17. Converta os seguintes AFNs em AFDs.



- b)  $\Sigma = \{a, b, c, d\}$ ; No autómato,  $\Sigma \setminus \{b\}$  representa todo o alfabeto  $\Sigma$  exceto o conjunto  $\{b\}$

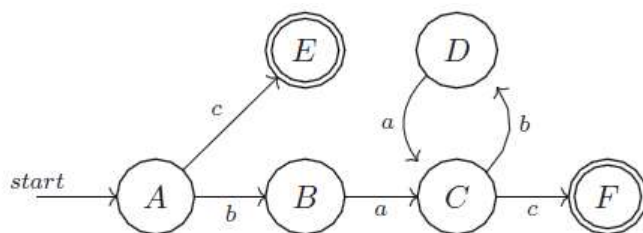


18. Minimize os seguintes autómatos.





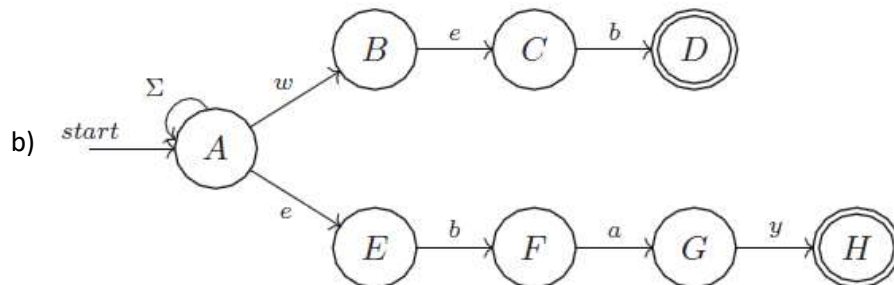
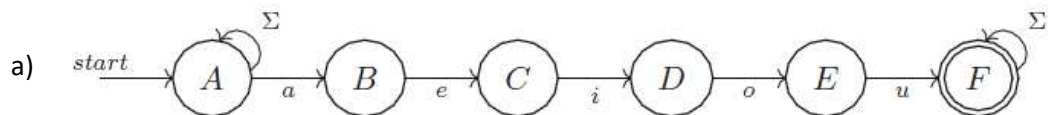
b)



19. Minimize o autómato representado pela seguinte tabela de transições:

	0	1
→S	A	F
A	C	B
B	F	A
C	A	F
*F	B	C

20. Considerando o alfabeto  $\Sigma = \{a, b, c, d, \dots, y, z\}$ , converta os seguintes AFNs com em AFDs.



21. Verifique se o autómato para reconhecimento de moedas criado no exercício 5, da secção 2, está minimizado.