

# Processamento de Linguagens

## Engenharia Informática (3º ano)

Teste (época normal)

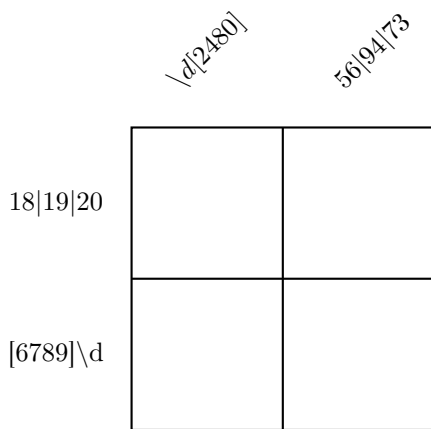
25 de Maio de 2024 (9h00)

Dispõe de **2:00 horas** para realizar este teste.

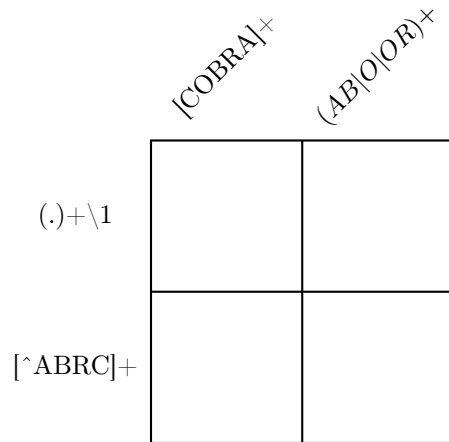
### Questão 1: Expressões Regulares (6v = 1.5+1.5+1.5+1.5)

A seguir são-te apresentados quatro matrizes com palavras cruzadas em que o enunciado são expressões regulares. Resolve cada um dos enigmas preenchendo a grelha com os caracteres devidos (na tua folha de resposta, coloca a grelha preenchida).

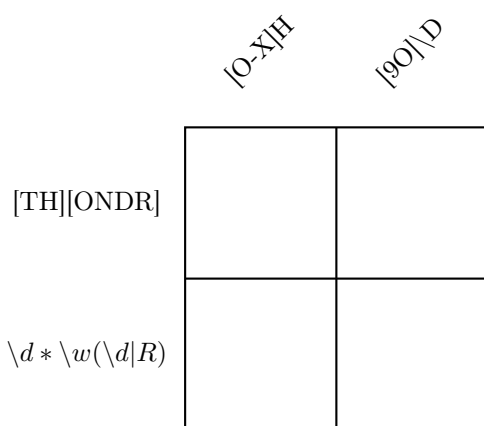
Obs: No enigma 4 tens de resolver as interseções das expressões regulares nos extremos de cada linha e coluna.



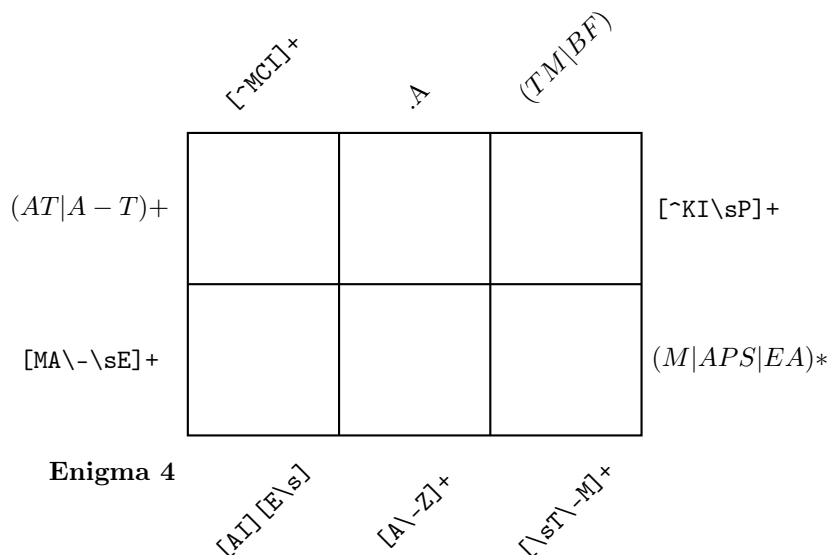
Enigma 1



Enigma 2



Enigma 3



Enigma 4

### Questão 2: filtros (4v = 1.5+1.5+1)

a) Considere a seguinte página HTML:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
```

```

4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Sample Form Page</title>
7  </head>
8  <body>
9      <header id="header">
10         <h1 id="header-title">Contact Us</h1>
11     </header>
12     <div id="form-container">
13         <form id="contact-form">
14             <div class="form-group" id="name-group">
15                 <label for="name" id="name-label">Name:</label>
16                 <input type="text" id="name" name="name">
17             </div>
18             <div class="form-group" id="email-group">
19                 <label for="email" id="email-label">Email:</label>
20                 <input type="email" id="email" name="email">
21             </div>
22             <div class="form-group" id="message-group">
23                 <label for="message" id="message-label">Message:</label>
24                 <textarea id="message" name="message" rows="4"></textarea>
25             </div>
26             <button type="submit" id="submit-btn">Submit</button>
27         </form>
28     </div>
29     <footer id="footer">
30         <p id="footer-text">&copy; 2024 Sample Form Page</p>
31     </footer>
32 </body>
33 </html>

```

Especifique um filtro em Python que extraia o valor de todos os atributos `id` e os coloque na saída ordenados alfabeticamente;

- b) Especifique um filtro em Python que indique quantas tags de início há na página (da forma: '`<identificador ...`') e quantas tags de fecho existem ('`</identificador>`').
- c) Mudando de contexto, considere um texto em que se tem um nome de uma pessoa por linha. Os nomes encontram-se capitalizados (primeira letra de cada palavra é maiúscula) e as palavras encontram-se separadas por um espaço branco.

```

Josefina Maria Carvalho Ramos
João Rui Carvalho
José Carlos Leite Ramalho
Pedro Rangel Henriques
Tiago João Fernandes Batista

```

Escreva uma função python usando expressões regulares, que dado este texto de input e uma sigla, por exemplo, `jcr` imprima os nomes completos que *condizem* com a sigla. Um nome condiz com a sigla se:

- a primeira letra da sigla é correspondente à primeira maiúscula do nome;
- as restantes letras da sigla deverão corresponder às outras maiúsculas do nome (não precisam de corresponder a todas, basta a um subconjunto mas terão de estar na mesma ordem).

Para a sigla `jcr`, o resultado seria:

```

Josefina Maria Carvalho Ramos
José Carlos Leite Ramalho

```

### Questão 3: Gramáticas ( $5v = 3+2$ )

Ao visitar um país, um viajante tem de visitar várias localidades e em cada uma delas tem de visitar vários pontos de interesse cultural para ver um monumento, uma exposição, um espetáculo, um panorama, etc. Essas visitas devem ser feitas por uma determinada ordem. Precisa também de registar o custo de cada atividade a realizar nesses vários pontos.

Neste exercício pede-se que:

- a) Imagine uma linguagem interessante para ajudar o viajante a fazer o seu plano de viagem por um ou mais países. Para definir essa linguagem, escreva, em BNF-puro, uma Gramática Independente de Contexto (GIC) apropriada;
- b) Identifique os símbolos terminais que usou na GIC da alínea anterior e escreva o respetivo analisador léxico associando a todos esses símbolos terminais as respetivas Expressões Regulares.

## Questão 4: Parser ( $5v = 1+1+1+1+1$ )

Considere o Terminal variável `pal` (sequência não nula de letras e hífens) e a seguinte Gramática Independente de Contexto (G) em que o axioma é `LingPaises` e `"&"` representa a string nula.

```

NT = { LingPaises, Pares, Par, Lingua, Paises, Pais }
T  = { '.,', ':', ', ', DIC, CID, ORIG, pal }
P  = {
    p1: LingPaises : DIC Pares CID
    p2: Pares      : Pares Par ' .'
    p3:           | &
    p4: Par        : Lingua ': ' Paises
    p5: Lingua     : pal ORIG pal
    p6: Paises     : Pais ', ' Paises
    p7:           | Pais
    p8: Pais       : pal
  }

```

Neste contexto e após analisar a  $G$  dada e a Linguagem  $L$  por ela definida, responda às alíneas seguintes:

- a) Considere a frase  $F$ :

```

DIC
  PT ORIG latim: Portugal, Brasil.
  FR ORIG latim: França.
CID

```

Apresentando a respetiva *árvore de derivação* mostre que  $F$  pertence a  $L$ .

Mostre ainda a diferença estrutural entre uma lista definida com recursividade à esquerda e uma lista definida com recursividade à direita.

- b) Os **Conflitos LL(1)** em  $G$  podem surgir nas produções p2/p3 ou nas produções p6/p7. Recorrendo ao *Lookahead* dessas produções, diga justificando se  $G$  é uma GIC do tipo **LL(1)**.
- c) Após estender a  $G$  dada, construa o respetivo *Autómato LR(0)*. Diga, justificando, se em algum desses estados ocorrem **Conflitos shift-reduce**.
- d) Suponha que  $G$  tinha sido modificada, mantendo  $L$ , da seguinte forma

```

P  = {
    ...
    p6: Paises      : Pais OutrosP
    p7: OutrosP     : ', ' Pais OutrosP
    p7':           | &
    p8: Pais        : pal
  }

```

Escreva, então, 2 funções de um parser RD (recursivo-descendente), para reconhecer os 2 símbolos não-terminais **Paises** e **OutrosP**.

- e) Acrescente Ações Semânticas às produções da gramática para calcular o número total de línguas diferentes registadas e outras Ações Semânticas para escrever o total de países em que se fala cada uma das línguas.