

# Teste teórico nº 1

---

- **Realizado na aula de 28/04**
- **Cotação: 7 valores**
- **Duração: 90 min, sem consulta**
- **Inscrição obrigatória: ver Moodle**
- **Tópicos**
  - Introdução à Integração de dados
  - Expressões regulares
  - XML / JSON
  - XPath
  - Validação com DTD

# Exercício 1 - ER

---

- Escreva uma expressão regular que encontre todas as palavras começadas e terminadas por vogais (maiúsculas e/ou minúsculas).
- Escreva uma expressão regular que encontre todas as palavras com 10 caracteres numéricos e/ou alfabéticos (maiúsculos ou minúsculos).
- Expressão regular que capture todas as palavras começadas e terminadas pela letra 'a' (ou 'A'), sendo o restante conteúdo letras minúsculas ou maiúsculas.
- Escreva uma ER que encontre palavras de tamanho 6 caracteres, começadas com três dígitos e terminadas com três vogais.
- Escreva uma ER que encontre cadeias binárias que sejam repetições da sequência 011
  - 011    011011011    0111011011011011

## Exercício 2 - ER

- Assinale as strings capturadas pelas seguintes ERs

$\backslash b(a^*b)^*\backslash b$

A -- ab  
B -- bbbb  
C -- ababa  
D -- abbba  
E -- ababab  
F -- aaaaaa  
G -- bababa

$\backslash b(a?b^*c^+)\{2,3\}\backslash b$

A -- ac  
B -- bbbc  
C -- abcabc  
D -- bcbcbc  
E -- bc  
F -- ccc  
G -- abccccbc

$\backslash b[01]?.[bc]^*\backslash b$

A -- 0Xbb  
B -- 1Xcbc  
C -- 11c  
D -- 01  
E -- 000cc  
F -- 01c  
G -- 11bc

# Exercício 3 - ER

- Assinale as strings capturadas pelas seguintes ERs

`\ba?b*s[0-9]*\b`

Select one or more:

- ☐ bbs
- ☐ s222
- ☐ ass333
- ☐ abs001
- ☐ aabs9

`\b[abc]?a[bc]*\b`

Select one or more:

- ☐ abc
- ☐ aaabc
- ☐ bccbc
- ☐ abbbbb
- ☐ ac

`^[test]+[ar]*\s[REre]{2}s?\s[diDI]+\? $`

Select one or more:

- ☐ esta re DI?
- ☐ test re di?
- ☐ es a re DI?
- ☐ testar er di?
- ☐ ta r di?

# Exercício 3 - XPath

---

- **Nomes (texto) dos concorrentes e dos jurados que vivem em Coimbra.**
- **Nomes dos concorrentes que moram em Lisboa e têm mais de 30 anos.**
- **Localidades (elementos cidade) onde moram os concorrentes com mais de 50 anos.**
- **Qual a idade do concorrente mais velho?**
- **Todos os concorrentes (elementos nome) que ficaram aprovados a mais do que uma categoria (cat).**
- **Quantos concorrentes foram aprovados a pelo menos duas categorias?**
- **Id do concorrente com a classificação mais alta à categoria (cat) 'teatro'**

## Exercício 3 - XPath

---

- **Transcreva do ficheiro o que é devolvido pela seguinte expressão XPath:**

**`/curso/descendant::concorrente[last()]/preceding-sibling::concorrente[position()<3]//cat[@tipo="dança"]//ancestor::*/nome`**

# Exercício 4 - DTD

---

- **Escreva o DTD para validar**
  - Elemento <concorrente>
  - Elemento <nome>
  - Atributo **resultado** como uma enumeração
  - Atributo **tipo** como string