

## Integração de Dados

Duração: 90 minutos

Licenciatura em Engenharia Informática: 2º ano - 2º semestre

2017/2018

### TESTE Nº 2

Nome: \_\_\_\_\_ Nº aluno: \_\_\_\_\_

#### 1. [20%] XSLT #1

Escreva o XSLT que permita transformar o ficheiro **escola1.xml** dado em anexo no seguinte ficheiro XML. O ficheiro contém todos os responsáveis ordenados por ordem alfabética do seu nome e o elemento **<num\_alunos>** contém o numero de alunos associado a cada responsável.

```
<listagem>
  <responsavel id="enc01">
    <nome>Anabela Costa</nome>
    <num_alunos>5</num_alunos>
  </responsavel>
  <responsavel id="enc03">
    <nome>Maria Matos</nome>
    <num_alunos>2</num_alunos>
  </responsavel>
  <responsavel id="enc02">
    <nome>Paulo Silva</nome>
    <num_alunos>2</num_alunos>
  </responsavel>
</listagem>
```

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:output method="xml" indent="yes"/>
  <xsl:template match="escola">
    <listagem>
      <xsl:apply-templates select="registo">
        <xsl:sort select="responsavel"/>
      </xsl:apply-templates>
    </listagem>
  </xsl:template>
  <xsl:template match="registo">
    <responsavel id="{responsavel/@id}" >
      <nome><xsl:value-of select="responsavel"/></nome>
      <num_alunos><xsl:value-of select="count(alunos/aluno)"/></num_alunos>
    </responsavel>
  </xsl:template>
</xsl:stylesheet>
```

## 2. [30%] XSLT #2

Escreva o XSLT que permita transformar o ficheiro **escola1.xml** dado em anexo no seguinte ficheiro HTML:

### Responsáveis de Coimbra

- Nome do Responsável: Anabela Costa
  - Nome do Aluno: Jorge Costa
  - Nome do Aluno: Laura Costa
  - Nome do Aluno: Maria Costa
  - Nome do Aluno: Nuno Costa
  - Nome do Aluno: Susana Costa
- Nome do Responsável: Paulo Silva
  - Nome do Aluno: Ana Silva
  - Nome do Aluno: Daniel Silva

O ficheiro contém a informação dos responsáveis de Coimbra, ordenadas por ordem alfabética de nome. Para cada responsável, aparece a lista de alunos, também ordenada alfabeticamente.

Na última coluna encontra-se o número de alunos de cada responsável.

Deve usar obrigatoriamente pelo menos uma instrução **for-each** e uma instrução **if** para a resolução deste exercício.

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html"/>
  <xsl:template match="escola">
    <html>
      <body>
        <h1>Responsáveis de Coimbra</h1>
        <ul>
          <xsl:for-each select="registo">
            <xsl:sort select="responsavel"/>
            <xsl:if test="local='Coimbra'">
              <li>Nome do Responsável: <xsl:value-of select="responsavel"/></li>
              <ul>
                <xsl:apply-templates select="alunos/aluno">
                  <xsl:sort select="@nome"/>
                </xsl:apply-templates>
              </ul>
            </xsl:if>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
  <xsl:template match="alunos/aluno">
    <li>Nome do Aluno: <xsl:value-of select="@nome"/></li>
  </xsl:template>
</xsl:stylesheet>
```

### 3. [20%] XQuery #1

Escreva uma expressão FLWOR em Xquery que permita fazer a mesma transformação da pergunta 1.

```
<listagem>{
  for $r in doc("escola1.xml")//registro
  order by $r/responsavel
  return
    <responsavel id="{ $r/responsavel/@id }">
      <nome>{ $r/responsavel/text() }</nome>
      <num_alunos>{ count($r/alunos/aluno) }</num_alunos>
    </responsavel>
}
</listagem>
```

### 4. [30%] XQuery #2

Escreva uma expressão FLWOR em Xquery que permita juntar a informação dos ficheiros **escola1.xml** e **escola2.xml** e crie o seguinte resultado em HTML:

#### Valores em dívida

Responsavel	Valor
Anabela Costa	400
Maria Matos	0
Paulo Silva	800

Prentende-se que, para cada responsável, seja mostrado o valor que tem em dívida (informação em **escola2.xml** no atributo “**pago**”x§). Esse total deve ser calculado por uma função de nome **calculaDívida** que recebe como argumento o identificador do responsável e devolve o somatório dos totais que essa pessoa deve.

Implemente a query que, usando a função anteriormente definida, produza o output HTML mostrado na figura acima. O resultado está ordenado por ordem alfabética do nome do responsável.

```
declare namespace xsd = "http://www.w3.org/2001/XMLSchema";
declare function local:divida($id as xsd:string?) as xsd:double?
{
  let $valor := sum(doc("escola2.xml")//aluno[@id_resp=$id and @pago="não"]/@valor)
  return $valor
};

<html><body>
<h2>Valores em dívida</h2>
<table border="1">
<tr><th>Responsavel</th><th>Valor</th></tr>
{
  for $r in doc("escola1.xml")//registro
  let $v := local:divida($r/responsavel/@id)
  order by $r/responsavel
  return <tr>
    <td>{ $r/responsavel/text() }</td>
    <td>{ $v }</td>
  </tr>
}
</table></body></html>
```