

Linguagens e Programação

Exame Época Normal

7 de julho de 2017

- Responda a **cada grupo em folhas separadas**. Deve entregar, pelo menos, **uma folha por grupo**
- Indique o **seu número e nome** em cada folha que entregar
- A **prova é com consulta** (1 folha A4) e tem a duração de 2 horas

Grupo I (6 valores)

1. [1,5 val.] Um programador que pretendia desenvolver uma função para verificar se havia números duplicados num vetor de inteiros, escreveu o seguinte código em Java.

```
public static boolean HasDuplicates (int [] v)
{
    for (int i = 0; i<=v.length; i++)
    {
        for (int j=0; j<=v.length();j++)
            if ((i <> j) and (v[i] = v[j]))
                return true;
    }
    return false;
}
```

O programador, ao compilar o programa que incluía esta função teve de corrigir vários erros e quando, finalmente, conseguiu executar o programa em modo de *debugging*, o programa terminou abruptamente, com a mensagem de erro: “*vector subscript out of range*”. Identifique, classifique e corrija todos os erros que o programador cometeu.

2. [1,5 val.] “ Diz-se que uma gramática é ambígua se existirem, pelo menos, duas frases a que corresponde uma árvore de derivação”. Comente a afirmação.
3. [1,5 val.] Numa definição dirigida pela sintaxe os atributos dos símbolos não terminais são somente herdados e fornecidos pelo analisador léxico”. Comente a afirmação.
4. [1,5 val.] Considere a seguinte gramática G:
- $$S \rightarrow iEtS \mid iEtSeS \mid a$$
- $$E \rightarrow b$$

Identifique as transformações necessárias na gramática G, por forma a construir um analisador sintático preditivo para a linguagem descrita.

Grupo II (4 valores)

1. [1 val.] Um clube de Hóquei pretende criar um sistema de automatização dos dados de um jogo. Especifique uma expressão regular que permita validar a informação referente a um jogo. Deve começar com o nome da equipa visitada seguida do número de golos marcados, o nome da equipa visitante seguida dos golos marcados, seguido de uma letra (F – se o numero de assistentes ao jogo for inferior ou igual a 1000, M – superior a 1000) e do número de pessoas a assistir ao jogo.

Considere que:

- o nome das equipas é composto por três letras maiúsculas
- o número de golos é composto por um ou dois dígitos (sem zeros à esquerda)
- o número de assistentes no jogo tem no máximo quatro dígitos (sem zeros à esquerda).

Exemplos:

SLB4FCP6F950

SCP5FCP12M1230

2. Considere o autómato finito A representado na Tabela 1:

Tabela 1

	0	1
→A	\emptyset	{B}
B	{D,C}	\emptyset
*C	{C}	{C}
D	\emptyset	{B}

- [0,5 val.] Classifique o autómato finito A. Justifique.
- [1,5 val.] Apresente o AFD minimizado equivalente ao autómato finito A, **usando o método formal de minimização**. Justifique todas as decisões e pressupostos que assumir.
- [0,5 val.] Represente graficamente o autómato finito minimizado, obtido na alínea b).
- [0,5 val.] Indique a expressão regular associada à linguagem aceite pelo autómato

Grupo III (4 valores)

1. Considere o alfabeto $\Sigma = \{a,b\}$:

- [0,5 val.] Defina as produções duma gramática regular capaz de reconhecer a linguagem cujas palavras são dadas pela seguinte expressão regular: **[ba]{2,}**
- [1,5 val.] Represente um Autómato Finito que reconheça a linguagem gerada por esta gramática.

2. Considere a seguinte gramática G:

$S \rightarrow TyBT$

$T \rightarrow xT|x$

$B \rightarrow zB|\epsilon$

- [0,5 val.] Classifique a gramática G, segundo a hierarquia de Chomsky. Justifique.
- [1 val.] A expressão regular $x+yz^*x+$ representa a mesma linguagem que a gramática G? Justifique.
- [0,5 val.] Apresente uma sequência de derivação mais à esquerda para a palavra **xyzxx**

Grupo IV (6 valores)

Considere a informação sobre um conjunto de animais tal como está no ficheiro `animais.xml`.

```
<?xml version="1.0" encoding="UTF-8" ?>
<animais>
  <cães>
    <cão>
      <nome>Bobi</nome>      <!-- todos os animais têm nome -->
      <nascimento>2016-03-05</nascimento> <!--podem não ter data de nascimento-->
    >
      <chip>123456789</chip>  <!-- todos os animais têm um chip de 9 dígitos -->
    >
      <raça>Rafeiro</raça>    <!-- apenas os cães têm raça -->
      <!-- raça pode ser um das seguintes: (Rafeiro,Rafeiro do Alentejo,Serra de
        Aires,Barbado da Terceira,Podengo) -->
    </cão>
  </cães>
  <gatos>
    <gato>
      <nome>Tareco</nome>
      <chip>335494555</chip>
    </gato>
    <gato>
      <nome>Amarelo</nome>
      <nascimento>2016-11-20</nascimento>
      <chip>655494555</chip>
    </gato>
  </gatos>
</animais>
```

- 1- [3 val.] Complete o esquema XSD apresentado de modo que permita validar o *XML*, sabendo que neste (tal como está no exemplo) só podem aparecer cães e gatos, por esta ordem. Defina os *complex types* e/ou *simple types* que considere necessários para completar o *schema*, tendo em atenção as restrições definidas no ficheiro XML acima apresentado.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="http://www.dei.isep.ipp.pt/animais"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.dei.isep.ipp.pt/animais"
  elementFormDefault="qualified">

  <xs:element name="animais" type="TAnimal"/>

  <!-- Completar schema -->

</xs:schema>
```

2. [3 val.] Completar o XSLT seguinte (**bloco A**, **bloco B** e **bloco C**) que produza duas tabelas com o nome e chip de cada animal. A primeira deve estar ordenada pelo nome e a segunda pelo chip. A transformação **não** pode usar a instrução “for-each”.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html"/>

  <xsl:template match="/">
    <html>
      <head/>
      <body>
        <h2 align="center">Lista de Nomes</h2>
        <table border="1" align="center">
          <tr>
            <!-- bloco A -->
          </tr>
        </table>

        <h2 align="center">Lista de Chips</h2>
        <table border="1" align="center">
          <tr>
            <!-- bloco B -->
          </tr>
        </table>
      </body>
    </html>
  </xsl:template>

  <!-- bloco C -->
</xsl:stylesheet>
```

Lista de Nomes

Nome:	Chip:
Amarelo	655494555
Bobi	123456789
Jeco do Monte Grande	963444555
Tareco	335494555

Lista de Chips

Chip:	Nome:
123456789	Bobi
335494555	Tareco
655494555	Amarelo
963444555	Jeco do Monte Grande