

# SPLN

## MI-EI

Exame, 02 de Fevereiro de 2018

**Questão 1 ()** Escreva da forma mais simples e elegante que conseguir:

- a). uma função perl que calcule a intercepção de multi-conjuntos:  $A \cap B$ ; Sugestão: represente os multi-conjuntos como HASH-references

`baginter( {a=>3,c=>3,e=>6},{ a=>2,c=>8,d=>9}) = {a=>2,c=>3}`

- b). uma função que dada um identificador em *Camel Case* possivelmente com sublinhados ou números (2 e 4) o separe os seus constituintes ("my\_idCompareSimple4sort" → "my id compare simple for sort")

---

**Questão 2 ()** Escreva script que faça a quebra de frases para textos (em Português):

- a). cada frase deve ficar numa única linha.  
b). as frases são terminadas por um ponto final.  
c). considere que há uma lista de abreviaturas (sr., dr., d. e prof.) cujo ponto não termina frase.  
d). se a seguir ao ponto surgir um letra minúscula, esse ponto não é um ponto final mas sim um ponto de abreviatura e deve ser ignorado como separador de frase.

---

**Questão 3 ()** Um dos modos habituais de distinguir entre duas línguas consiste em calcular previamente a tabela de probabilidade dos trigramas de caracteres e *aplicar* essa tabela ao texto em análise.

1	#trigrama	pt	en
2	mat	1.962	2.257
3	the	0.002	20.323
4	ing	1.352	15.267
5	ent	3.024	2.809
6	çõe	2.443	0.000
7	anh	1.443	0.001
8	...		

- a). Faça programa ou que dado um corpus de uma língua, calcule a coluna da tabela de trigramas correspondente (os números devem estar em ocorrências por milhão).  
b). Faça programas que dado um ficheiro tabtrig (com colunas PT e EN) e dado um texto, indique a que língua pertence.
-

```

%%
z → json EOF
;
json → str
    | '[' lista ']'
    | '{' mapping '}'
;
lista → json
    | lista ',' json
;
mapping → str ':' json
    | mapping ',' str ':' json
;
%%

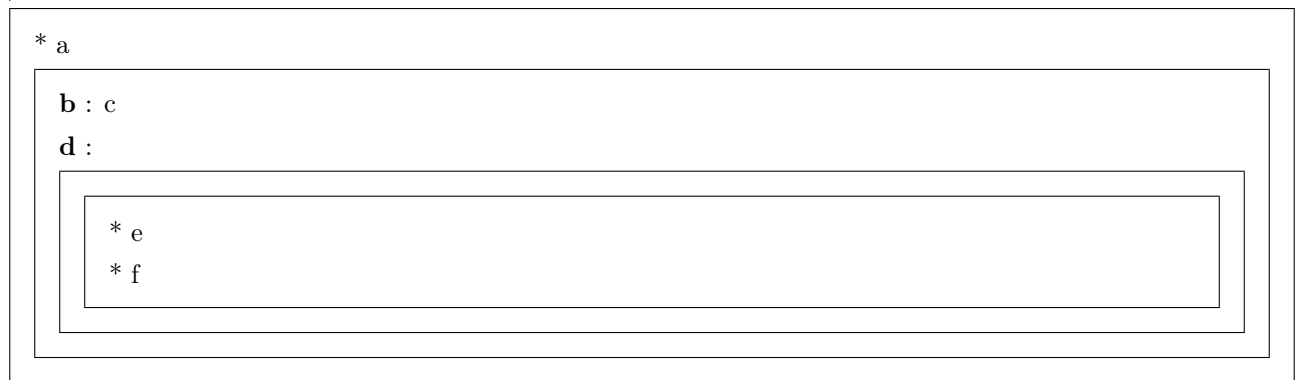
```

```
["a",{ "b": "c", "d": ["e", "f"]}]
↓
\begin{framed}
  * a

  \begin{framed}
    \textbf{b} : c

    \textbf{d} :
      \begin{framed}
        \begin{framed}
          * e

          * f
        \end{framed}
      \end{framed}
    \end{framed}
  \end{framed}
\end{framed}
```



**Questão 6 (P)** ara ajudar a estudos sobre a língua portuguesa, foram disponibilizados grandes quantidades de textos anotados linguisticamente, seguindo o seguinte formato:

```

1 <meta>
2 jornal publico
3 </meta>
4 <s>
5 Estes     este     DET 0   P   M   >N 0
6 temas    tema     N   0   00 0   SUBJ> 0
7 serão    ser V     FUT 3P 0   FS-STA 0
8 retomados retomar V     PCP P   M   ICL-AUX< 0
9 na em+o   PRP+DET 0   S   F   <ADVL+>N 0
10 próxima próximo ADJ 0   S   F   >N 0
11 reunião reunião N   0   00 0   P< 0
12 de de PRP 0   0   0   N< 0
13 de de PRP 0   0   0   N< 0
14 Câmara câmara N 0   00 0   P< 0
15 do de+o   PRP 0   S   M   N< +>N 0
16 distrito distrito N 0   00 0   P< 0
17 de de PRP 0   0   0   N< 0
18 Leiria Leiria PROP 0   00 0   P< 0
19 , , PU 0   0   0   PONT 0
20 a a PRP 0   0   0   <ADVL 0
21 realizar realizar V   INF 3   0   ICL-P< 0
22 no em+o   PRP+DET 0   S   M   <ADVL+>N 0
23 dia dia N 0   00 0   P< 0
24 de de PRP 0   0   0   N< 0
25 Abril abril N 0   00 0   P< 0
26 , , PU 0   0   0   PONT 0
27 nas em+o   PRP+DET 0   P   F   <ADVL+>N 0
28 Caldas Caldas PROP 0   00 0   P< 0
29 . . PU 0   0   0   PONT 0
30 </s>
31 <s>
32 ...

```

Onde as colunas estão separadas por um tab (\t) e contêm respectivamente: palavra, lema, categoria gramatical, etc.

- a). Escreva uma script que calcule quantas vezes ocorreu cada par (lema-categoria) num texto. Exemplo de saída esperada:

```

1 este-DET      83
2 tema-N       23
3 ...

```

- b). Desenvolva um Filtro que reconstrua o texto original, imprimindo apenas o conteúdo do *meta* e a primeira coluna das *s* de acordo com o seguinte formato:

```

1 ==jornal publico
2 1. Estes temas serão retomados na ..... , nas Caldas .
3 2. A câmara de .....

```