

# Estruturas de Dados e Algoritmos II — 1ª Frequência

Departamento de Informática — Universidade de Évora

10 de Abril de 2015

1. [3,5 valores] Assumindo que o alfabeto consiste nas 26 letras maiúsculas, desenhe uma *trie* cujo conteúdo sejam as cinco palavras

CAML ML C COBOL OCAML

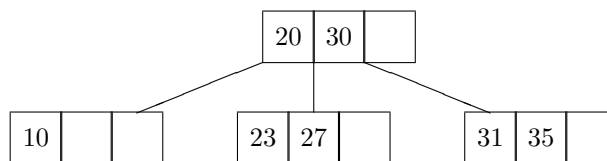
Qual seria a memória ocupada por uma implementação em C da *trie* que desenhou, numa máquina com endereços e palavras de 32 bits? (Não precisa de calcular o valor, mas apresente e justifique todos os cálculos efectuados ou a efectuar.)

2. [2,5 valores] Que campo se poderia incluir nos nós de uma *trie* para tornar mais eficiente apagar os nós da *trie* após a remoção de uma palavra? Como é que esse novo campo ajudaria?

3. [3,5 valores] A *B-tree* da figura tem grau de ramificação mínimo 2. Apresente o seu estado depois da execução de *cada uma* das operações da sequência

r 30 r 35 r 20 r 31 i 15 i 3 i 22

pela ordem apresentada. As letras **i** e **r** indicam, respectivamente, a inserção e a remoção do elemento que se lhes segue.



4. [3 valores] Pretende-se armazenar numa *B-tree* registos constituídos por: três campos de texto, com até 50 caracteres cada; e quatro inteiros, com valores até 10 000.

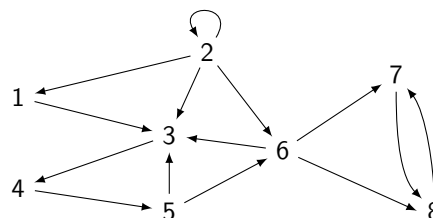
Se a *B-tree* for implementada num sistema com páginas de 4096 *bytes*, qual será o grau de ramificação mínimo apropriado para a árvore? Explícite e justifique todas as decisões tomadas. (Não é necessário chegar ao valor final, basta apresentar o resultado sob a forma de uma expressão cujo valor seja a resposta.)

5. [3,5 valores] Apresente o pseudo-código da função B-TREE-REMOVE-MAX(x), que remove e devolve o maior elemento da *B-tree* cuja raiz é o nó x (que tem ocupação superior à mínima), e analise a sua complexidade temporal.

Considere que os nós de uma *B-tree* têm os campos *n* (ocupação), *c* (filhos), *key* (elementos) e *leaf* (é folha?).

Pode assumir que as funções B-TREE-BORROW-FROM-LEFT-SIBLING(x, i) (que implementa o empréstimo de um elemento de  $x.c[i-1]$  a  $x.c[i]$ ) e B-TREE-MERGE-CHILDREN(x, i) (que funde  $x.c[i]$  e  $x.c[i+1]$ ) estão definidas.

6. Considere o grafo orientado representado na figura.



- (a) [1 valor] Apresente um caminho simples de comprimento 5.
- (b) [1 valor] Apresente um ciclo simples de comprimento 4.
- (c) [1 valor] Diga, justificando, se o grafo é fortemente conexo.
- (d) [1 valor] Apresente as componentes fortemente conexas do grafo.