

Pesquisa e Classificação de Dados

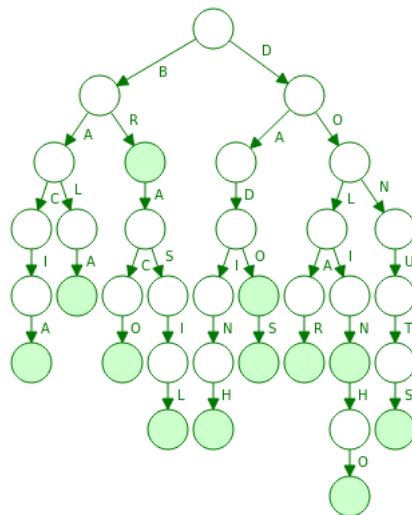
Lista 4 (Tries/Radix/Patricia, Huffman)

Prof. Ricardo Oliveira

Esta lista **não** vale nota e **não** deve ser entregue, mas apenas utilizada como material de apoio para estudo. Naturalmente, você pode tirar eventuais dúvidas com o professor.

Exercícios marcados com (B) são básicos e essenciais para a matéria. Exercícios marcados com (C) são complementares. Recomenda-se fortemente a resolver todos os exercícios.

1. (B) Em uma trie inicialmente vazia, insira, nesta ordem, as chaves SINAL, CAO, CAL, PAPO, C, XISTUDO, X, CALOI, XIS, S, XISTO, PAPA, PA e SIM, e apresente a trie resultante.
2. (B) Considerando a trie criada no exercício anterior:
 - (a) Remova SINAL e indique quantos nodos foram removidos;
 - (b) Após, remova XISTO e indique quantos nodos foram removidos;
 - (c) Após, remova XIS e indique quantos nodos foram removidos;
 - (d) Após, remova XISTUDO e indique quantos nodos foram removidos;
 - (e) Após, remova X e indique quantos nodos foram removidos;
 - (f) Após, remova CALOI e indique quantos nodos foram removidos;
 - (g) Após, remova PA e indique quantos nodos foram removidos;
 - (h) Apresente a trie resultante.
3. (B) Considere a seguinte trie:



- (a) Apresente, em ordem, todas as chaves com prefixo DO;
- (b) Apresente, em ordem, todas as chaves com prefixo B;
- (c) Apresente, em ordem, todas as chaves com prefixo DADI;

- (d) Apresente, em ordem, todas as chaves com prefixo DADA;
 - (e) Apresente, em ordem, todas as chaves.
4. (C) Seja Σ um alfabeto qualquer (não necessariamente $\Sigma = \{A, B, \dots, Z\}$). Considere uma trie de N chaves com no máximo M letras (de Σ) cada.
 - (a) Apresente um limitante superior para o número de nodos na trie.
 - (b) Considere que cada nodo tem um vetor de $|\Sigma|$ ponteiros para seus filhos. Indique um limitante superior para a complexidade de espaço da trie.
 - (c) Determine *exatamente* o número máximo de nodos que a trie pode ter considerando que sua altura é um inteiro h dado.
 5. (C) Considere que, ao invés de um vetor com $|\Sigma|$ ponteiros, cada nodo contém uma árvore AVL com uma chave para cada filho que o nodo tem. Analise a complexidade de tempo da busca e inserção de uma chave de tamanho M nesta trie.
 6. (B) Apresente um algoritmo recursivo que imprime todas as chaves contidas em uma trie, em ordem. Utilize uma pilha como estrutura auxiliar.
 7. (C) Utilizando uma trie como estrutura auxiliar, apresente um algoritmo para ordenar um vetor de strings.
 8. (C) Analise a complexidade de tempo do algoritmo apresentado no exercício anterior.
 9. (C) Apresente algoritmos iterativos (não recursivos) para inserção e busca em uma trie com a mesma complexidade de tempo dos algoritmos recursivos estudados.
 10. (B) Apresente a árvore radix contendo as chaves listadas no exercício 1.
 11. (B) Na árvore radix obtida no exercício anterior, insira, nesta ordem, as chaves SINOS, PAPIRO, CA, VIDA e XISTUDINHO. Apresente a árvore resultante.
 12. (B) Da árvore radix obtida no exercício anterior, remova, nesta ordem, as chaves CAL, PA, CA e XIS. Apresente a árvore resultante.
 13. (B) Apresente a árvore digital contendo os seguintes inteiros com quatro bits: 9, 13, 11, 0, 3, 5 e 15.
 14. (B) Apresente a árvore PATRICIA contendo as chaves listadas no exercício anterior.
 15. (B) Considere um arquivo de texto contendo **SIESINDDASESSENUNDWIRSINDDIEJAGER**. Apresente sua tabela de frequência, uma árvore de Huffman, seu código, e o arquivo compactado. Compare o tamanho dos dois arquivos.
 16. (B) O código de Huffman também pode ser construído, de maneira semelhante, dada uma tabela de *percentual de frequência esperada*, isto é, para cada letra, qual a porcentagem esperada de ocorrências daquela letra. Com isso, é possível criar um único código para compactar e descompactar vários arquivos (por exemplo, de um mesmo idioma). Considerando a seguinte tabela de frequência esperada: S=24% A=30% P=21% R=10% O=15% (e considerando que, durante a construção da árvore, o filho de um nó de menor (resp. maior) frequência recebe rótulo 0 (resp. 1)):
 - (a) Determine seu código de Huffman;
 - (b) Compacte POROS;
 - (c) Descompacte 10011001010111.