

Departamento de Informática - UFPR  
Algoritmos e Estruturas de Dados III - CI057 - 2017/1  
Segunda prova  
Prof. Eduardo Almeida e Prof. Marços Didonet

- A prova tem duração de 1h30m.
- A interpretação faz parte da prova. Pode fazer a lápis.
- PROVA SEM CONSULTA.

- (20pts) 1. Defina uma estrutura de dados para armazenar uma árvore 'Trie', com o alfabeto de A até Z [A-Z]. Em seguida, defina um algoritmo que receba um prefixo como entrada, percorra a árvore e que imprima todas as palavras com este mesmo prefixo.
- (30pts) 2. Um 8-heap é um heap onde cada elemento pode ter até 8 filhos. A representação em vetor segue uma estrutura semelhante aquela usada no heap binário. Na 8-heap se a raiz da estrutura ocupa a posição 1 do vetor, as próximas 8 posições, de 2 até 9, são ocupadas pelos elementos que são filhos da raiz. A mesma lógica segue recursivamente para os filhos. A função  $Filho(i, n)$  retorna o índice do  $n$ -ésimo filho do nó de índice  $i$ . Por exemplo  $Filho(1, 1) = 2$ ,  $Filho(1, 8) = 9$ ,  $Filho(3, 2) = 19$ . Escreva um algoritmo em C ou em pseudocódigo semelhante a C que implemente a função  $Filho(i, n)$  para um 8-heap.
- (20pts) 3. Comprima o texto "abracadabra mostre a cabra", sem as aspas, utilizando o algoritmo de Huffman apresentado em sala. Você deve apresentar a árvore binária gerada e a quantidade de bits originais e comprimidos. Os espaços em branco devem ser considerados como último caracter na ordem alfabética.
- (30pts) 4. Realize as seguintes operações e apresente as árvores B e B+ resultantes (com  $t = 2$ ): incluir (12, 29, 7, 20, 13, 2, 5), excluir(29, 12), na sequência apresentada. A exclusão deverá ser feita pela chave antecessora.