

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

- (30pts) 1. Um 4-heap é um heap onde cada elemento pode ter até 4 filhos. A representação em vetor segue uma estrutura semelhante aquela usada no heap binário. Na 4-heap se a raiz da estrutura ocupa a posição 1 do vetor, as próximas 4 posições, de 2 até 5, 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 $Filho(1, 1) = 2, Filho(1, 4) = 5, Filho(3, 2) = 11..$ Escreva um algoritmo em C ou em pseudocódigo semelhante a C que implemente a função $Filho(i, n)$ para um 4-heap.
- (20pts) 2. Em uma tabela hash com 100 entradas, as colisões são resolvidas usando listas encadeadas. Para reduzir o tempo de busca, decidiu-se que cada lista seria reorganizada como uma árvore binária de busca. A função utilizada é $h(k) = k \bmod 100$. Infelizmente, as chaves inseridas seguem o padrão $k_i = 50i$, onde k_i , corresponde à i -ésima chave inserida. Mostre a situação da tabela após a inserção de k_i , com $i = 1, 2, 3, \dots, 13$. Faça o desenho da tabela resultante.
- (20pts) 3. Utilizando o algoritmo de Huffman apresentado em sala, compactar o texto "bom esse bom bom" e apresentar a árvore binária gerada e o código binário de compressão do texto. Os espaços em branco devem ser considerados como último caracter na ordem alfabética.
- (30pts) 4. Dada a árvore vermelho-preto abaixo, realize as seguintes operações e apresente a árvore válida resultante: incluir (90,41,40,25,14), excluir(27,29,40,41,42). Os nodos pretos são representados por $\langle p \rangle$, enquanto que os nodos vermelhos são representados por $\langle v \rangle$. A exclusão deverá ser feita pela chave sucessora.

