

## Exercícios de Revisão para Prova Final (SEM ENTREGA)

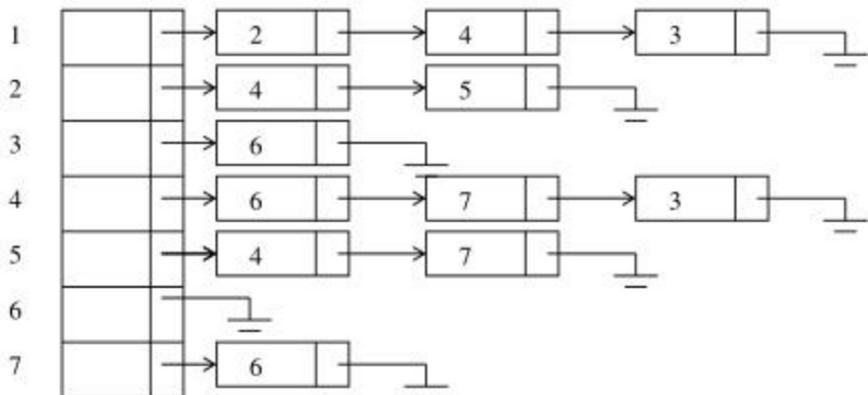
### Parte I – Exercícios de Implementação

Para os exercícios a seguir descompacte os arquivos “Revisao.zip”.

**Obs:** Faça uso das boas práticas de programação: nomenclatura intuitiva das classes, métodos, atributos e variáveis além do uso do encapsulamento, o desleixo com essas práticas será penalizado na Avaliação Final

1. Crie uma classe com um método estático que receba como parâmetro uma Lista (**objeto da classe contida no pacote**) e remova os elementos múltiplo de 10. Além disso, os valores excluídos devem ser inseridos em uma pilha nativa do JAVA do tipo: Stack<Integer>. Ao final do processo, a pilha deve ser retornada da função.

2. Crie uma classe com tabela Hash indireta com objetos da lista flexível (**Lista implementada no pacote**) que possua os métodos: construtor com o parâmetro nº de elementos da tabela Hash, inserir elemento, pesquisar elemento e remover elemento. A figura a seguir ilustra os dados na Hash.



3. Implemente no interior da classe AVL (**classe contida no pacote**) um método que retorne a mediana dos valores contidos na árvore.

**OBS:** Para encontrar o valor da mediana é necessário colocar os valores em ordem crescente ou decrescente. Quando o número de elementos da árvore for par, a mediana é encontrada pela média dos dois valores centrais. Outrora é ímpar, a mediana é o valor do elemento central. Pelo menos um passo deste algoritmo deve ser recursivo.

### Parte I – Exercícios Manuscritos

4. Seja a sequência de números {100, 52, 71, 16, 33, 57, 62, 36, 7, 87, 69 e 10} respectivamente. Ilustre o passo-a-passo do método de:

- a) inserção na árvore AVL, apresente os fatores de平衡amento, operações de rotação;

2	[1,0]	$R_E$
	-1	$R_{DE}$
-2	[-1,0]	$R_D$
	1	$R_{ED}$

- b) inserção na árvore tipo 2-3-4, com tipo de fragmentação a sua escolha.

5. Seja a sequência {5, 6, 2, 10, 9, 1 e 8}, ilustre o passo de inserção nas seguintes *hashs*.

- a) Hash direta com área de reserva (overflow). Suponha uma tabela de tamanho 9 com uma área reserva igual a 2.

Função hash(int x){return x%9}

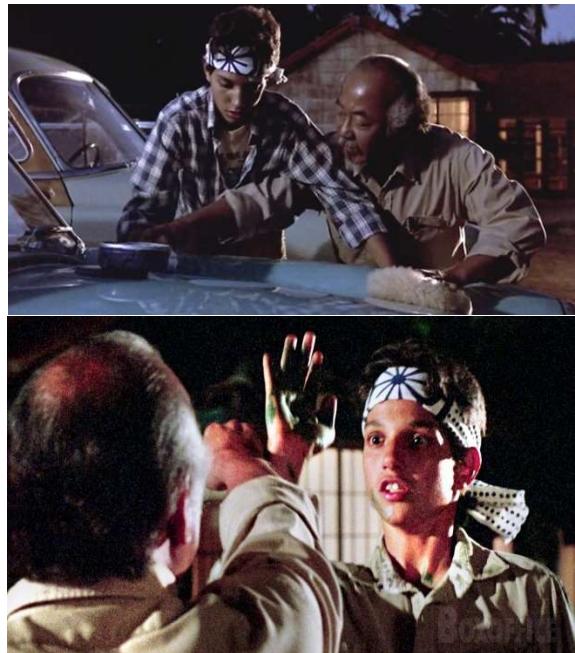
- b) Hash direta com Rehash com tamanho 9.

Função hash(int x){return x%9}

Função rehash(int x){return ++ x%9}

- c) Hash indireta com lista flexível com tamanho 5.

### Para descontrair



“Mostre-me passando cera, tirando cera!”

