

1) Fazer o teste de mesa conforme o algoritmo abaixo. O teste de mesa deve deixar claro o console e a estrutura de dados ao final da execução.

Vetor:

42	46	16	-9	48	20	35	37	11	44	43	12	20	48	45	45	31	21	47	4	20	42	44	-7	3	36
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	----	----	----	----	---	----

```
Lista l = new Lista();
Para (int valor : vetor) {
    Se (listaVazia()) {
        l.addFirst(valor * 2);
    } Senao Se (l.size() == 1) {
        l.addFirst(valor);
    } Senao Se (valor < 0) {
        l.addLast(valor);
    } Senao Se (valor % 3 == 0) {
        l.add(valor / 3, 1);
    } Senao {
        l.add(valor * valor, 1);
    }
}

int tamanho = l.size();
enquanto(! l.isEmpty()) {
    Se (tamanho == 1) {
        escreva(l.get(0));
        l.removeFirst();
    } Senao Se (tamanho > 10) {
        escreva(l.get(0));
        l.removeFirst();
    } Senao {
        escreva(l.get(tamanho - 1));
        l.removeLast();
    }
    tamanho = l.size();
}
```

2) Considere os valores abaixo que devem ser armazenados em um vetor de 10 posições usando Tabela de Espalhamento pelo método do Encadeamento.

A função hash é dada baseada no método da multiplicação.

Faça um teste de mesa de como ficará a inserção dos valores na tabela, deixando claro, ao final, como todos os dados estarão dispostos na tabela.

Entrada:

30	23	17	12	76	58	98	58	24	36	92	17	75	74	78	78	0	11	67	17
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	----	----	----

Função Hash:

$$f = (\text{int})(10 * (((7 / 13) * \text{valor}) \text{MOD } 1))$$

*A operação número MOD 1 remove a parte inteira do número, deixando só a parte decimal

3) Considere uma lista de telefones no formato String (XXNNNNNNNNNN) onde XX é o DDD e o NNNNNNNNNN é o número de celular, criar uma função hash que coloque qualquer número de telefone em um vetor que começa no endereço 11 e termina no 99. Pode ser usando Pseudocódigo ou Java. A função hash deve retornar um número inteiro.

4) Considere um número grande de entrada de valores numéricos restritos ao intervalo de 0 a 99. Criar uma função hash que disponha esses valores em um vetor de 10 posições com a seguinte característica:

Posição	Dados
0	Valores de 0 a 9
1	Valores de 10 a 19
2	Valores de 20 a 29
3	Valores de 30 a 39
4	Valores de 40 a 49
5	Valores de 50 a 59
6	Valores de 60 a 69
7	Valores de 70 a 79
8	Valores de 80 a 89
9	Valores de 90 a 99

5) Montar uma árvore conforme a representação em parênteses abaixo:

(15 (25 (39 (105) (250)) (50 (40) (98)) (4 (40)))

6) Montar uma árvore de busca binária, baseada em suas regras, a partir dos dados abaixo, apresentar como ficará o atravessamento dessa árvore por pré ordem, em ordem e pós ordem:

27	68	60	28	191	3	21	130	121	66	16	48	9	190	146	160	24	182	26	0
----	----	----	----	-----	---	----	-----	-----	----	----	----	---	-----	-----	-----	----	-----	----	---

7) Considerando a árvore de busca binária abaixo, responder:

- Como resultará o conjunto de saída do atravessamento em ordem
- Qual a altura dessa árvore?
- Qual o nível do dado 10 ?
- Como ficará a nova árvore, aplicando uma estratégia para a remoção do número 4?

