

Exercícios sobre Listas Simples e Duplamente Ligadas

Escreva os métodos para a Classe Lista e para a Classe Node:

- 1) Retorne a quantidade de nós da lista (length)
- 2) Retorne a soma dos valores armazenados na lista (total)
- 3) Retorne o maior valor armazenado na lista (max)
- 4) Retorne o menor valor armazenado na lista (min)
- 5) Escreva o método doubleX() que deve dobrar os valores armazenados na lista.
- 6) Recebe um valor inteiro (valor) como parâmetro e o insere no início da lista (addInicio).
- 7) Escreva o método int somaPosImpares(), que fará a soma dos valores dos nós nas posições ímpares da lista encadeada. Considere que o Início está na posição 0, como se fosse um vetor.
- 8) Escreva o método void posPares() que deve exibir na tela os nós das posições pares da lista.
- 9) Recebe um valor inteiro (pos) como parâmetro e retorna o valor armazenado no nó cuja posição seja igual a pos. Considere que o início da lista ocupa a posição 0.
- 10) Recebe um valor inteiro (pos) como parâmetro e remove o nó da lista que corresponde a essa posição. Caso essa posição não exista, deve exibir na tela uma mensagem de erro: "Posição não existe"
- 11) Construa uma variação para o método do exercício 10 que retorna o valor do nó retirado da lista. Caso a posição não exista, deve retornar o valor -999999.
- 12) Construa os métodos inserePos(posição, valor) que faça a inserção do valor na posição indicada por posição. Note que se a posição for posterior ao último nó da lista, o método deve inserir esse valor como se fosse o último. Caso a posição seja 0, deve inserir o valor como se fosse o primeiro nó da lista e nos demais casos, deve inserir entre dois nós, refazendo as ligações do novo nó com os nós anterior e posterior.