

Exercício de LAB II - Listas Duplamente Encadeadas

Prof. João Valiati

Faça uma classe que é filha da classe DLinkedList (lista duplamente encadeada), nela implemente os seguintes métodos:

- a) um método que permita inserir um novo nodo com novo conteúdo depois do nodo que contenha um conteúdo desejado. Exemplo: `addAfter("4","3")`.
- b) um método que encontre o nodo com o conteúdo de maior valor e faça a remoção desse nodo, fazendo o reposicionamento adequado da lista. Exemplo: `removeMax()`.
- c) um método que recebe uma posição (inteira) e remove o nodo nessa posição, fazendo o reposicionamento adequado da lista. Exemplo: `removePos(3)`.
- d) no método `main` dessa classe crie uma nova lista de `String` e realize os seguintes testes fazendo uso dos respectivos métodos implementados anteriormente. Inserção dos valores "2", "4", "6", "8". Inserção do "3" após o "2", do "5" após o "4" e do "9" após o "8". Inserção do "7" antes do "8". Remoção do nodo na posição 3 e remoção do nodo com o maior valor. Entre cada atividade de inserção e remoção imprima a lista na tela para verificação.

Dica: no exercício descrito no item "b" para encontrar o nodo com maior conteúdo você deve varrer a lista para encontrar o nó de maior conteúdo, para fazer isso você deve utilizar o `compareTo`. Devido a uma restrição da máquina virtual java a sua classe deve estender `Comparable`. Da seguinte forma:

```
public class MinhaClasse<E extends Comparable<E>> extends DLinkedList<E>{ }
```

Entrega: Via Moodle da disciplina. Data final de entrega: (18/11/2016 até às 22:23 hs). O código deve estar dentro de um pacote Java chamado "NomeSobrenomeExerc12".

Compactar apenas os arquivos fontes (com extensão .java) e enviar.