

[ED193] Removendo todas as ocorrências de um elemento

Neste problema deverá apenas submeter uma classe **SinglyLinkedList<T>** (e não um programa completo).

Código Base

Use como base a classe **SinglyLinkedList<T>** ([ver código](#) | download de [Node.Java](#) e [SinglyLinkedList.Java](#)), que representa uma lista ligada simples e tem disponíveis métodos para adicionar ou remover um elemento no início ou no final, devolver o tamanho, saber se a lista está vazia ou retornar representação em *string* para escrita (tal como dado nas aulas).

O problema

Acrescente à classe dada um novo método **public void removeAll(T value)** que **remove da lista todas as ocorrências do valor *value***. Note que para comparar os elementos deverá usar o método **equals** e não um simples == (pode assumir que os objectos da lista têm implementada uma versão correcta do **equals**).

Submissão

Deverá submeter apenas a classe **SinglyLinkedList<T>**, acrescentando o método **removeAll** como pedido (e **sem apagar nenhum dos outros métodos dados como base**). Pode assumir que terá acesso no Mooshak à classe Node<T> (não a pode mudar) e se precisar pode criar outros métodos auxiliares. O Mooshak irá criar várias instâncias da sua classe e irá fazer uma série de testes ao método por si implementado.

Exemplos de Input/Output

Lista inicial	Chamada	Estado da lista depois da chamada
list = {1,2,2,2,1,3,4,2,1}	list.remove(1)	list = {2,2,2,3,4,2}
list = {1,2,2,2,1,3,4,2,1}	list.remove(2)	list = {1,1,3,4,1}
list = {1,2,2,2,1,3,4,2,1}	list.remove(3)	list = {1,2,2,2,1,4,2,1}
list = {1,2,2,2,1,3,4,2,1}	list.remove(5)	list = {1,2,2,2,1,3,4,2,1}
list = {"cc","cc","cc","cc","cc"}	list.remove("cc")	list={}

