

Para efeitos da nota atribuída à resolução de exercícios ao longo do semestre - **Submeter até 23:59 de 12 de Junho**
(o problema continuará depois disponível para submissão, mas sem contar para a nota)
[para perceber o contexto do problema deve [ler o guião da aula #12](#)]

[ED205] Árvores estritamente binárias

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

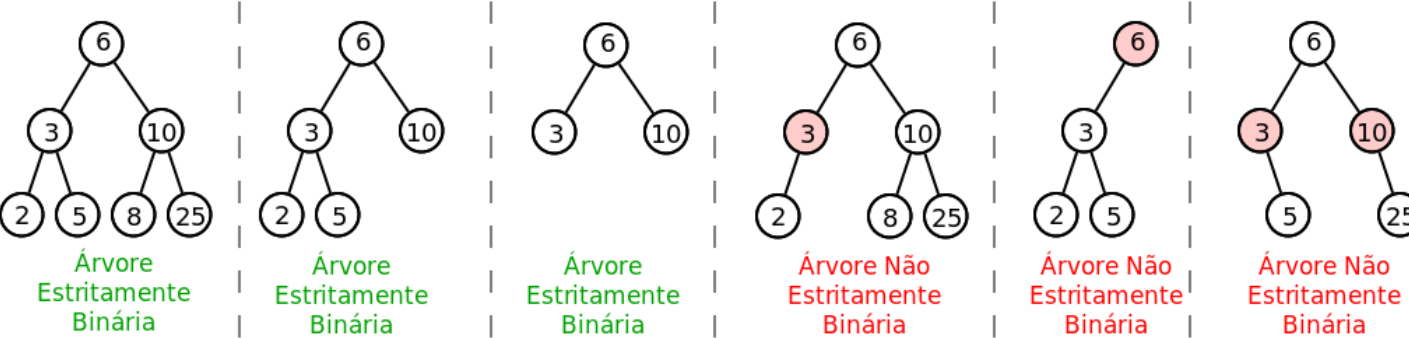
Código Base

Use como base a classe **BTree<T>** ([ver código](#) | download de [BTNode.Java](#) e [BTree.Java](#)), que representa uma árvore binária, tal como dada nas aulas).

O problema

Acrescente à classe dada um novo método **public boolean strict()** que **devolve true** se a **árvore for estritamente binária** ou **false** caso contrário..

Uma árvore é **estritamente binária** se não existir nenhum nó só com um filho, ou seja, se todos os seus nós têm exactamente dois filhos ou são folhas (zero filhos). A figura seguinte ilustra seis árvores binárias, sendo que as três primeiras são estritamente binárias e as três últimas não (a ver vermelho estão indicados os nós que não respeitam a condição pedida).



Submissão

Deverá submeter apenas a classe **BTree<T>**, acrescentando o método **strict** como pedido (**e sem apagar ou modificar nenhum dos outros métodos dados como base**). Pode assumir que terá acesso no Mooshak à classe **BTNode<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

Os exemplos correspondem às seis árvores da figura:

Árvore em preorder	Chamada	Valor de retorno
t = 6 3 2 N N 5 N N 10 8 N N 25 N N	t.strict()	true
t = 6 3 2 N N 5 N N 10 N N	t.strict()	true
t = 6 3 N N 10 N N	t.strict()	true
t = 6 3 2 N N N 10 8 N N 25 N N	t.strict()	false
t = 6 3 2 N N 5 N N N	t.strict()	false
t = 6 3 N 5 N N 10 N 25 N N	t.strict()	false