

1. Adicione à interface `Map<K, V>` o método *size*, que retorna a quantidade de chaves inseridas na coleção.
2. Adicione à interface `OrderedMap<K extends Comparable<K>, V>` os seguintes métodos:
  - `rank(K val)`: retorna a quantidade de chaves da coleção menores ou iguais a *val*;
  - `size(K lo, K hi)`: retorna a quantidade de chaves com valores entre *lo* e *hi*.
3. Na classe `BinarySearchTreeMap<K extends Comparable<K>, V>`, implemente os novos métodos definidos nas interfaces. Dica: na classe `Node`, adicione o campo *count*, para armazenar a quantidade de chaves da subárvore cujo nó é raiz. Ajuste os métodos de inserção e remoção para atualizar esse valor.
4. Mostre que uma árvore binária completa com  $n$  nós tem altura  $\log_2^{n+1} - 1$ .
5. Na classe `BinarySearchTreeMap`, escreva um método para retornar a altura da árvore.

Divirta-se!