



Lista - Generics-Collections

1) Escreva uma classe genérica que recebe um tipo T qualquer. A classe terá como atributos dois valores do tipo T, e possuirá os seguintes métodos:

- getMax

- toString: retorna uma string com a concatenação dos dois valores, com o de maior valor na frente

a) resolva usando o método `double doubleValue()` herdado da classe `Number`; teste objetos de classes boxing (`Double`, `Long`, `Integer`, etc);

b) resolva usando o método `int compareTo(T o)` definido na Interface `Comparable<T>`; teste com uma classe `Produto(String nome, double preco)`.

2)

a) Escreva o método de ordenação insertion-sort* de maneira genérica. Use o seguinte algoritmo:

`Insertion-sort(TIPO[] A)` /*Não use ArrayList, apenas array simples*/

 Laço $j \leftarrow 1 < A.\text{tamanho}$

 temp $\leftarrow A[j]$

$i \leftarrow j-1$

 Enquanto $i \geq 0$ E $A[i] > \text{temp}$

$A[i+1] \leftarrow A[i]$

$i \leftarrow i-1$

 Fim Enquanto

$A[i+1] \leftarrow \text{temp}$

 Fim Laço

Fim

➔ Modifique o algoritmo para que ele use o método `compareTo` ao invés do operador `>`. Teste seu algoritmo com uma classe boxing, como `Integer` ou `Float`.

b) Escreva uma classe `Time Futebol` que implementa a interface `Comparable` implementando o método `compareTo`.

c) Escreva um método principal que usa a classe do item b) com o algoritmo do item a). Execute o método sobre um array com 10 elementos, imprimindo os dados antes e depois da ordenação.

*O insertion-sort é um algoritmo de ordenação por comparação que tem péssimo desempenho esperado de $O(n^2)$; ele só deve ser usado em razão de sua simplicidade, e em operações de ordenação com poucos elementos.

3) Escreva uma classe que define uma árvore binária de busca genérica recebendo um parâmetro T. Sua árvore deverá ter métodos de inserção, busca, e impressão em ordem de seus valores. ATENÇÃO: Não é necessário mantê-la balanceada, o que torna a implementação mais simplificada. Teste com a classe `Integer`.

4) Escreva uma Pilha Genérica para trabalhar com qualquer tipo de dados. Ela deve se basear em um array primitivo do tipo T e possuir os métodos `bool empilhar(T)` e `T desempilhar()`. Teste com `Strings`.

5) Implemente uma calculadora não genérica. Ela deve possuir um método estático que calcula a média a partir de uma lista genérica de elementos que possuam o método `double doubleValue()`.