

Introdução à Linguagem Java

Lista 08

Instruções

Nesta aula, o uso de ferramentas de busca como o Google, a documentação do Java¹, sites sobre programação como o StackOverflow, ou mesmo o Wikipedia será essencial, pois a ideia é vocês pesquisarem como resolver um problema. Por isso, antes de tentarem resolver algum exercício, procurem um pouco e vejam como resolvê-lo de maneira correta

Exercício 1 (4,0)

Usando a biblioteca Java Swing² do Java, crie uma interface simples para o programa de Fatorial criado nas aulas anteriores. Basta colocar um lugar onde o usuário possa entrar um número, um botão para iniciar o cálculo e um lugar para mostrar o resultado. Separem o código da interface (a *View*) do código que faz o cálculo de Fatorial (o *Controller*). Não esqueçam do tratamento de exceções, o programa de vocês não deve retornar resultados errados nem fazer o usuário se deparar com uma mensagem de erro esquisita!

Exercício 2 (2,0)

Como fazer testes automatizados do exercício anterior? Qual a diferença desse tipo de teste com os testes usando o JUnit que estávamos fazendo até agora? Dica: procurem sobre testes de unidade e testes de aceitação. Fazer um teste “de aceitação” parcialmente automatizado e explicar, em um comentário, como um usuário deve proceder.

Exercício 3 (2,0)

Crie quatro classes que representam multiconjuntos de elementos quaisquer, ou seja, conjuntos em que um elemento pode ocorrer mais de uma vez.

Você pode criar classes desses objetos usando classes existentes nas bibliotecas java. Você pode usar herança de classes existentes, ou usá-las como

¹<https://docs.oracle.com/javase/7/docs/api/>

²https://en.wikipedia.org/wiki/Swing_%28Java\%29

membros. Crie 4 classes de objetos desse tipo através de `ArrayList`, `Set`, `LinkedList` e `Stack`.

Em cada caso, implemente os métodos `add(T element)`, `equals(Multiconjunto<T> m)`, `addAll(Multiconjunto<T> m)`.

Faça testes automatizados que evidenciem o funcionamento de cada estrutura de dados e suas possibilidades, mostrando como ocorre a “junção” de duas ou mais estruturas, dentre outras funcionalidades.

Exercício 4 (2,0)

Procure na internet sobre o Padrão Estrutural de projetos **Adaptor**. Crie um adaptador que permite que o `Multiconjunto` do exercício anterior seja utilizado como um `Conjunto` (ignorando as repetições). Use iteradores para percorrer o `multiconjunto` e insira um iterador no adaptador para que o `multiconjunto` possa ser percorrido adaptadamente como um `conjunto`.

Exercício extra (+1,0)

Crie um programa que mostre a diferença de desempenho entre `String` e `StringBuffer` na operação de concatenação de duas strings. Dê uma breve explicação de por que isso ocorre. Dica: a classe `String` é imutável, enquanto a classe `StringBuffer` é mutável.