

### **LISTA 03: HERANÇA (AULA 05)**

1. Faça primeiramente os exercícios do capítulo 21 da apostila (Kon, F.; Goldman, A.; Silva e Silva, P.; Introdução à Ciência da Computação com Java e Orientação a Objetos, IME-USP, São Paulo, 2005.).
2. Qual a diferença entre o mecanismo de herança e interfaces?
3. Revise os papéis do Teatro de Objetos visto em nossa primeira aula. Quais papéis poderiam ser implementados em forma de hierarquia de classes (herança)? Modele essas classes em UML.
4. Implemente as classes do exercício anterior (dependendo da ação necessária, os métodos podem chamar métodos de outras classes e/ou imprimir mensagens como “Jogador camisa x bateu o pênalti”).
5. Imagine que uma empresa tenha produtos dos quais precisa armazenar o código, o nome e o preço. No entanto, os produtos perecíveis exigem que sejam armazenadas também a data de validade (por enquanto use um número inteiro) e o nome do último fornecedor (String). Faça uma hierarquia de classes (com atributos e métodos *get* e *set*) para essas classes. Os dados do produto devem ser fornecidos quando o objeto for criado (o mesmo ocorre para produtos perecíveis). As classes criadas devem ter métodos para imprimir dados, exercitando o reuso de código.
6. Você foi contratado para compor uma estrutura de classes para armazenar os alunos da USP. Faça uma superclasse com os atributos: número USP, nome do aluno e nome do curso. Em seguida, você deve representar duas situações: aluno que faz iniciação científica (atributos: valor da bolsa (double), orientador, órgão financiador da bolsa, e nome do projeto – todos do tipo string) e aluno que faz estágio (atributos nome da empresa - string, valor da bolsa – double, data de início - inteiro e data de finalização - inteiro). Faça uma estrutura de classes usando herança para proporcionar o máximo possível de reuso de código. Faça métodos para obter, alterar e imprimir todos os atributos.