

Gabarito da 2ª Lista de exercícios de Linguagens de Programação (DCC 206)
Turma 01 (2020.1)

1. A maioria das linguagens imperativas não possui classes e métodos. Como você usaria uma linguagem imperativa como Pascal ou C para simular as ideias de classes e métodos?

R: Criando cada estrutura abstrata (que irá simular uma classe) e suas respectivas funções (que irão simular os métodos) em arquivos separados e, em seguida, criando um arquivo com o código principal onde as "classes" serão importadas.

2. A maioria das linguagens atuais não dá suporte à herança múltipla diretamente. Tente elaborar pelo menos dois problemas nos quais a herança múltipla é útil.

R:

Problema 1:

Quando temos uma classe "relógio" (com os atributos hora, minuto e segundo) e uma classe "calendário" (com os atributos dia, mês e ano) sendo herdadas por uma classe "agenda".

Problema 2:

Quando temos uma classe "gerente" (que autoriza vendas) e uma classe "vendedor" (que processa pedidos) sendo herdadas por uma classe "gerente de vendas".

3. Ainda usando uma linguagem imperativa como C, como você simularia o mecanismo de herança? É possível simular herança múltipla nessa linguagem?

R: Criando estruturas de estruturas; sim, é possível simular herança múltipla nessa linguagem.

4. Ada possui formas de definir tipos abstratos. Discuta porque ela não é considerada uma linguagem orientada a objetos.

R: Apenas possuir formas de definir tipos abstratos não torna uma linguagem de programação orientada a objetos. Contudo, a partir de 2006 a linguagem ADA passou a incluir os demais conceitos de orientação a objetos. Inclusive, é a primeira linguagem de programação orientada a objetos padronizada internacionalmente.

5. Argumente, sob o ponto de vista prático, porque as linguagens orientadas a objetos se tornaram tão populares e no que elas favorecem a reutilização.

R: Porque elas facilitam a abstração da realidade e como são baseadas nos princípios de herança e polimorfismo facilitam a implementação e reutilização de código.