

EXERCÍCIO 3 – EX3

Assunto:

Sobrecarga, Construtor, Reuso de Classe.

Orientações:

A atividade deve ser feita de forma **individual**. Pesquisem as respostas em livros, apostilas ou na Internet. As respostas (ver regras abaixo) devem ser enviadas para o e-mail jefferson@lesc.ufc.br com o assunto **[POO 2017.2] – EX3** até o final da aula do dia 22/12.

Regras de criação dos programas:

Crie um novo projeto Java no Eclipse denominado **Exercicio3**. Nesse projeto deve conter as seguintes classes encapsuladas com suas devidas funcionalidades abaixo:

Classe Disciplina

Deve ter:

1. Os atributos **Nome**, **Carga Horária** (representando o total de horas a serem cursadas) e **Semestre** (a que semestre do curso a disciplina pertence);
2. Um Construtor que exige **Nome**, **Carga Horária** e **Semestre** e atribui para seus atributos respectivos;
3. Outras versões do Construtor que possam omitir **Carga Horária** (valor padrão = 80) e **Semestre** (valor padrão = 1). Assim, se o usuário omitir a carga horária, por exemplo, a classe já sabe que se trata de uma disciplina de 80 horas;
4. Método **toString()** para imprimir suas informações completas e deve ter um método **getCargaHoraria()** para que se possa solicitar o valor de carga horária da disciplina.

Classe Usuario

Deve ter:

1. Os atributos **Nome**, **Matrícula** e **Carga Horária Total**;
2. Um atributo estático chamado **Numero de Cadastros**. Esse valor deve ser incrementado sempre que um novo Usuario for instanciado (partindo do zero). Naturalmente, deve-se criar um método para exibir quantos Usuários foram instanciados;
3. Um **Construtor** que exige **todos** os atributos (**exceto o atributo estático**, que inicializa e se modifica independente do que acontece com Usuário);
4. Método **toString()** que imprime todas as suas informações.

Classe Aluno

Essa classe é um tipo específico de Usuário e deve ter, em adição ao contido em Usuário:

1. Três atributos Disciplina, D1, D2 e D3 (é aceitável usar conjunto de Disciplinas, como vetor ou ArrayList, você escolhe);
2. Método **Matricular**, que recebe **Indice**, **Nome**, **Carga Horária** e **Semestre** e instancia uma nova Disciplina no Índice correspondente. Por exemplo: **Matricular(2,"Cálculo",40,3)** vai instanciar uma nova disciplina em D2 (ou no segundo elemento do vetor) chamada "Cálculo", com 40h de carga horária, no semestre 3;

3. Sobrecargas para o Método **Matricular**, de modo que ainda seja obrigatório os argumentos **Índice** e **Nome**. Por exemplo: `Matricular(3,"Circuitos Elétricos")` instancia em D3 uma disciplina de Circuitos Elétricos com 80h de carga horária e do semestre 1;
4. Método **CalcularCargaHorária()** que não recebe argumentos e ativa o `getCargaHoraria()` das disciplinas instanciadas para calcular a carga horária total do Aluno. Esse valor é salvo no atributo `Carga Horária Total`;
5. **Construtor** que recebe obrigatoriamente **Nome** e **Matrícula**, apenas (**não** passa Carga Horária, deve ser inicializada com zero);
6. Método **toString()** deve exibir as informações contidas em Aluno, incluindo os atributos de Usuário e as informações de suas Disciplinas.

Classe Teste

Essa classe deve conter o método `main` que realiza o seguinte procedimento:

1. Instancia um Usuario U chamado José, matrícula 102 e 40h de carga horária;
2. Instancia um Aluno A chamado João, matrícula 85;
3. Matricular A nas disciplinas (use as sobrecargas quando aplicável):
 - a. Cálculo, 80h, Semestre 1;
 - b. POO, 60h, Semestre 5;
 - c. Física, 40h, Semestre 1;
4. Chamar o método `CalcularCargaHorária` de Aluno para atualizar o total de horas cursadas.
5. Exibir as Informações de U e de A;
6. Exibir quantos Usuários foram instanciados.