

ESQUEMA AULA PRÁTICA 4

- ❑ Definição de classes instanciáveis
- ❑ Protecção de variáveis de instância
- ❑ Métodos construtores
- ❑ Métodos de consulta e modificação de variáveis de instância
- ❑ O método toString

1 - Pretende-se implementar a classe Contador que foi estudada nas aulas teóricas.

a) As instâncias da classe Contador deverão representar contadores do tipo inteiro capazes de responder a um conjunto de mensagens que implementam as seguintes operações:

- criar um Contador com valor inicial igual a zero;
- criar um Contador com valor inicial igual a uma valor dado pelo utilizador;
- incrementar o Contador de uma unidade;
- incrementar o Contador de uma valor dado pelo utilizador;
- decrementar o Contador de uma unidade;
- decrementar o Contador de uma valor dado pelo utilizador;
- consultar o valor do Contador ;

- implemente ainda o método **toString**.

b) Construa uma classe de Teste para verificar a correcção da classe anterior.

c) Depois de testar o método toString estude o que acontece quando numa instrução de escrita coloca apenas o nome do objecto sem lhe enviar a mensagem toString. Finalmente, transforme em comentário o método toString e volte a executar o programa anterior. O que acontece?

d) Construa um programa que gere aleatoriamente N valores inteiros no intervalo de $[-100, 100[$ sendo o valor de N dado pelo utilizador. Usando objectos da classe Contador o programa deverá contar quantos valores gerados são positivos e quantos são negativos.

2 – Pretende-se uma classe que represente os Empregados de uma empresa. Um Empregado tem um número de segurança social, um nome e um salário.

a) Defina o cabeçalho e os atributos;

b) Defina o construtor de omissão;

- c) Defina o construtor de omissão;
- d) Defina um construtor que receba como parâmetro o número de segurança social e o nome do empregado;
- e) Defina os métodos de consulta (getters) e os de modificação (setters);
- f) Defina o método toString;
- g) Construa ainda um método que permita subir o salário do empregado de uma dada percentagem dada como parâmetro.

3 – Construa uma classe de teste que permita verificar todos os métodos da classe Empregado.

4 – Construa um programa que permita “gerir” os empregados de uma empresa.

Deverá ter as opções:

- 1** – Criar empregado;
- 2** – Consultar todos os empregados;
- 3** – Modificar um empregado;
- 4** – Apagar um empregado.

5 – Suponha uma classe que representa as 3 cadeiras mais críticas de um curso.

a) Construa a classe CadeirasCriticas, definindo os atributos *curso* (*uma String*) que representa o nome do curso, e *disciplinas* (*um array com 3 Strings*) que representa as 3 disciplinas críticas desse curso. Defina os atributos, um construtor que receba como parâmetro o nome do curso, construa os getters os setters e o método toString.

b) Construa uma classe para testar **cada um dos métodos** da classe CadeirasCriticas. Deverá criar pelo menos dois objectos do tipo cadeiras críticas, instanciar os objectos, atribuir valores a todos os atributos do objectos, modificar esses valores e mostrar o conteúdo de cada objeto.

d) Teste novamente o método setDisciplinas, declarando um array de Strings com dimensão 3 (de nome novasCriticas), inicializando-o com valores à sua escolha, e usando-o para modificar o valor da lista de cadeiras criticas do objeto. No final invoque o método toString para ver o conteúdo do objeto.

De seguida atribua novos valores ao array novasCriticas, no main da classe de teste. Invoque o método toString e analise o que aconteceu? O seu método setDisciplinas estava mesmo correto? Se não estava, corrija-o.