

Laboratório I

Exercício sobre Herança e Polimorfismo (1)

Questão 1. Crie uma classe Aluno, que possui um nome e uma idade. Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso. Sobrescreva o método toString.

Questão 2. Crie uma classe chamada Cor, que possui 3 inteiros, referentes à quantidade de R, G e B da cor. Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso. Sobrescreva o método toString.

Questão 3. Crie uma classe chamada Data. Uma data possui 3 inteiros, referentes ao dia, mês e ano. Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso. Sobrescreva o método toString, que retorna a data no formato "dia/mês/ano".

Questão 4. Crie uma classe chamada Cadeira, que possui uma cor (objeto do tipo Cor) e um atributo do tipo Aluno, que indica o aluno que está ocupando a cadeira. Sabendo que quando uma cadeira é criada ela não está ocupada, crie um construtor que recebe apenas a cor da cadeira por parâmetro. Crie os métodos de acesso. Sobrescreva o método toString.

Questão 5. Crie uma classe chamada CadeiraComMesa, que é um tipo de cadeira e representa uma cadeira com mesa acoplada. Esta classe possui dois atributos, que representam a largura e a profundidade da mesa acoplada. Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso necessários. Sobrescreva o método toString.

Questão 6. Crie uma classe chamada CadeiraComRodinhas, que é um tipo de cadeira e representa uma cadeira com rodinhas nos pés. Esta classe possui um atributo que representa a data da última manutenção desta cadeira (objeto do tipo Data). Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso necessários. Sobrescreva o método toString.

Questão 7. Crie uma classe chamada Sala, que possui um nome e 4 cadeiras (do tipo Cadeira). Crie um construtor que recebe parâmetros para inicializar todos os atributos. Crie os métodos de acesso necessários. Sobrescreva o método toString.

Questão 8. Na classe Sala, crie um método que retorna a quantidade de cadeiras com mesa que estão livres para alguém sentar, ou seja, que não estão ocupadas por algum aluno. Chame este método de cadeirasComMesaDisponiveis. Lembre-se de que as 4 cadeiras que existem na sala podem ser cadeiras com rodinhas e/ou cadeiras com mesas acopladas.

Questão 9. Na classe Sala, crie um método que retorna a quantidade de cadeiras com rodinhas que estão livres para alguém sentar, ou seja, que não estão ocupadas por algum aluno. Chame este método de cadeirasComRodinhasDisponiveis. Lembre-se de que as 4 cadeiras que existem na sala podem ser cadeiras com rodinhas e/ou cadeiras com mesas acopladas.

Questão 10 (2.0pt). Na classe sala, crie um método chamado entraAluno, que recebe um objeto do tipo Aluno e uma String que representa o tipo de cadeira que ele quer sentar por parâmetro. O parâmetro do tipo de cadeira terá dois valores: “mesa” ou “rodinha”. Assim sendo, o método deve colocar o aluno recebido por parâmetro em uma cadeira do tipo desejado. Caso não tenha cadeira do tipo desejado disponível, o aluno deve sentar em uma cadeira de outro tipo que estiver disponível. O método retorna a cadeira que o aluno sentou ou null caso não haja cadeiras disponíveis.

Questão 11. Crie uma classe chamada Principal. Nesta classe, crie o método main. No main, faça o que se pede:

- Crie um objeto do tipo Sala, com todos os parâmetros necessários sendo recebidos pelo Teclado
- Imprima as informações da sala
- Tente fazer 6 alunos entrarem na sala criada. As informações a serem solicitadas devem ser obtidas a partir de leituras do Teclado. A cada inserção, imprima na tela o que aconteceu com a tentativa de entrada do aluno na sala: se ele entrou (ou seja, ocupou uma cadeira), imprima as informações da cadeira que ele ocupou; se ele não entrou, imprima uma mensagem informando que não havia lugar na sala.
- Imprima a área de todas as mesas acopladas nas cadeiras que possuem mesa
- Imprima a data da última manutenção de cada cadeira com rodinhas presente na sala
- Insira as seguintes linhas NAS PRIMEIRAS LINHAS da classe Principal:

```
import javax.swing.*;  
import java.awt.Color;
```

Peça para o Usuário digitar o nome de um aluno para procurar na sala. Em seguida, utilize o código abaixo (no final do seu método main) para abrir uma janela contendo a cor da cadeira que este aluno está sentado (caso o aluno seja encontrado em alguma das cadeiras da sala):

```
JFrame a = new JFrame();  
a.setSize(200, 200);  
a.setBackground(new Color(R, G, B));  
a.setVisible(true);
```

Atente para o fato de que no trecho de código “a.setBackground(new Color(R, G, B));”, as letras R, G e B devem ser substituídas pelo conteúdo respectivo da cor da cadeira que o aluno está sentado. Se tudo der certo, uma janela com a cor da cadeira vai ser aberta.