



Prova de Programação Orientada a Objetos – POO

Prof. Pedro Carlos da Silva Lara

Noite

Instituto Superior de Tecnologia em Ciência da Computação de Petrópolis

Nome: _____

Data: ____ \ ____ \ ____

Questão 1) [2,5 pts.] Crie uma classe denominada **Impressora**, esta classe possui os seguintes atributos: velocidade (ppm *paper per minute*), tipo de impressão (laser ou jato de tinta), tipo de cor (colorido ou preto e branco), marca. Crie um construtor que inicialize cada atributo (crie apenas um construtor). Crie os métodos **get** e **set** para cada atributo. Crie um método denominado **tempoDeImpressao** que receberá o número de folhas e retorna o tempo de impressão em minutos. Implemente apenas o método **tempoDeImpressao**.

Questão 2) [2,5 pts.] Crie uma classe denominada **Elevador** para armazenar as informações de um elevador dentro de um prédio. A classe deve armazenar o andar atual (0=térreo), total de andares no prédio, excluindo o térreo, capacidade do elevador, e quantas pessoas estão presentes nele. O construtor deve receber como parâmetros: a capacidade do elevador e o total de andares no prédio. A classe deve também disponibilizar os seguintes métodos:

```
entra          // para acrescentar uma pessoa no elevador (só deve acrescentar se
                // ainda houver espaço).
sai            // para remover uma pessoa do elevador (só deve remover se
                // houver alguém dentro dele).
sobe           // para subir um andar (não deve subir se já estiver no último andar).
desce          // para descer um andar (não deve descer se já estiver no térreo).
get....        // métodos para obter cada um dos os dados armazenados.
```

Obs: implemente apenas os métodos **entra** e **sobe**.

Questão 3)[2,5 pts.] Crie uma classe abstrata denominada **ContaBancaria** esta classe terá um atributo do tipo **float** denominado **saldo** que contém o saldo desta conta. Esta classe também terá um atributo que identifica o tipo de conta e o número da agência e o número da conta. Crie métodos para manipular o saldo da conta (**sacar** e **depositar**). O saldo sempre se inicia com R\$0,00. O método virtual puro será **printInfo** que imprime todas as informações da conta. Defina apenas um construtor que receberá o tipo da conta bancária, o número da agência e o número da conta. Crie duas classes concretas: **ContaPoupanca** e **ContaCorrente** que herdam de **ContaBancaria**. A classe **ContaPoupanca** terá um atributo **float** que indica o rendimento mensal da conta. Crie os construtores adequados. Obs: não precisa implementar nenhum método.

Questão 4)[2,5 pts.] Usando o conceito de polimorfismo de classes e o exercício anterior crie uma classe denominada **Cliente** que possui uma conta bancária e um nome. Crie e **implemente** um método denominado **sacar** que irá retirar um determinado valor da conta que o cliente possui. Observe que não será necessário conhecer o tipo da conta previamente. Crie um método denominado **depositar** que receberá um valor que será acrescentado na sua conta. Crie os gets e sets e construtores adequados para essa classe. Obs: só precisa implementar o método **sacar**.