**1)** Implemente uma classe chamada **Pessoa** que armazena informações como nome e endereço. Seu construtor deve receber esses dois atributos com parâmetros. A classe deve possuir métodos getters e setters para os atributos citados anteriormente. Além disto, ela deve possuir um método "imprimirInformacoes" que imprime o nome e o endereço armazenado na classe no formato "nome:endereço", sem as aspas. Duas classes **Estudante** e **Professor** herdam da classe anterior.

Para estudantes, é necessário armazenar os cursos feitos e as respectivas notas. Implemente um método chamado "adicionarCursoNota" que adiciona um curso junto com a nota correspondente, os cursos e as notas são ambos armazenados em um vector. A classe possui métodos getters para obter as notas e os cursos, um método "getNotaMedia" que calcula e retorna a média dos alunos, e um método "imprimirNotas" que imprime as notas dos estudantes.

Para os professores, deve-se armazenar os cursos ensinados atualmente, sendo possível  adicionar ou remover um curso. As operações de adicionar um curso não adicionam um curso repetido. Um atributo booleano que indica se a operção foi realizada ou não é retornado nas operações de adição e remoção.

O curso é representado apenas por um valor alfa-numérico (string).

**2)** mundo das formas é muito rico. Considerando isso, implemente uma hierarquia de classes que deve ter a classe básica Forma, a partir da qual as classes FormaBidimensional e FormaTridimensional são derivadas.

A classe Forma deve possuir dois atributos, centroX e centroY, que indicam o centro da figura no plano cartesiano. Uma figura tridimensional possui adicionalmente o atributo centroZ, responsável por indicar a coordenada no eixo z. Além diso, toda Forma possui um método "imprimir", responsável por imprimir o nome da forma (primeira letra em maiúsculo) seguido das coordenadas cartesianas entre parênteses, sem espaços, por exemplo, "Esfera(4,1,7)". Finalmente, caso a coordenada de algum dos eixos não seja fornecida, a mesma deve ser inicializada, por padrão, em 0.

Cada FormaBidimensional deve conter uma função "getArea" para calcular a área da respectiva forma bidimensional. Cada FormaTridimensional deve ter as funções "getArea" e "getVolume" para calcular a área da superfície e o volume do sólido, respectivamente.

Obrigatoriamente devem ser implementadas (adicionadas na hierarquia) as formas: Circulo, Quadrado, Cubo e Esfera. No caso de Circulo/Esfera, o primeiro parâmetro do construtor específico representa o raio. No caso de Quadrado/Cubo, o primeiro parâmetro do construtor específico representa o comprimento do lado.

**3)** Os serviços dos Correios oferecem várias opções de entrega, cada uma com custos específicos. Crie uma hierarquia de classes para representar diferentes tipos de entregas de pacotes disponíveis.

Crie uma classe Pacote como a classe básica da hierarquia, e inclua as classes PacoteDoisDias e PacoteNoturno que derivam de Pacote.

A classe básica Pacote deve possuir atributos que representam nome, endereço, cidade, estado e CEP tanto do remetente como do destinatário do pacote, além de atributos que armazenam o peso (em quilos) e o custo por quilo para a entrega do pacote. O construtor da classe Pacote é responsável pela inicialização dos atributos. Garanta que o peso e o custo por quilo sejam valores positivos ou zero. A classe Pacote também deve fornecer um método público "calcularCusto" que retorna o custo associado com a entrega do pacote multiplicando o peso e o custo por quilo.

A classe PacoteDoisDias é filha da classe Pacote, mas inclui um atributo que representa uma taxa fixa que a empresa de entrega cobra pelo serviço de entrega em dois dias. O construtor da classe PacoteDoisDias recebe um parâmetro para inicializar esse atributo. PacoteDoisDias també deve redefinir o método calcularCusto para que ele calcule o custo de entrega adicionando a taxa fixa ao custo baseado em peso calculado pelo método calcularCusto da classe Pacote.

A classe PacoteNoturno também herda da classe Pacote e contém um atributo adicional para representar uma taxa adicional por quilo cobrado pelo serviço de entrega noturno. PacoteNoturno deve redefinir a função calcularCusto para que ela acrescente a taxa adicional por quilo ao custo padrão por quilo antes de calcular o custo total da entrega.