

**1. Defina Orientação à Objetos, apresentando os conceitos de classe, objeto e enfatizando o papel da troca de mensagens. (2 pontos)**

R: Programação Orientada a Objetos é um modelo de análise, projeto e programação de software baseado na composição e interação entre diversas unidades chamadas de objetos. Os objetos são operados com o conceito de 'this' (isto) ou 'self' (si), de forma que seus métodos modifiquem os dados da própria instância. Os programas são arquitetados através de objetos que interagem entre si. Uma classe é uma estrutura que abstrai um conjunto de objetos com características similares. Uma classe define o comportamento de seus objetos - através de métodos - e os estados possíveis destes objetos - através de atributos.

Em outras palavras, uma classe descreve os serviços oferecidos por seus objetos e quais informações eles podem armazenar. Uma classe representa um conjunto de objetos com características afins. Uma classe define o comportamento dos objetos através de seus métodos, e quais estados ele é capaz de manter através de seus atributos. Um objeto, na vida real, é qualquer coisa a qual pudermos dar um nome. Um objeto é capaz de armazenar estados através de seus atributos e reagir a mensagens enviadas a ele, assim como se relacionar e enviar mensagens a outros objetos.

Atributos são características de um objeto. Basicamente a estrutura de dados que vai representar a classe. Exemplos: um objeto da classe "Funcionário" teria como atributos "nome", "endereço", "telefone", "CPF", etc. Sob o ponto de vista da programação orientada a objetos, um objeto não é muito diferente de uma variável normal. Um programa orientado a objetos é composto por um conjunto de objetos que interagem através de "trocas de mensagens". Na prática, essa troca de mensagem traduz-se na aplicação de métodos a objetos.

**2. Você está implementando um sistema de controle de vendas de uma loja de departamentos.**

- a. Defina uma classe Java para representar os tipos de produtos vendidos na loja e outra para os itens disponíveis no estoque. (1,5 pontos)**

ProdutosDisponiveis

ProdutosEstoque

- b. Considere-se escrevendo o método main. Escreva o código para instanciar produtos e itens de estoque definidos no item anterior. (1,5 pontos)**

```
class ProdutosDisponiveis{  
    string roupa;  
    void setRoupa(string nRoupa) {
```

```

        this.roupa = nRoupa;
    }
    string getRoupa() {
        return roupa;
    }
}
class ProdutosEstoque{
    string bolsa;
    void setBolsa(string nBolsa) {
        this.bolsa = nBolsa;
    }
    string getBolsa() {
        return bolsa;
    }
}
}
pra instanciar:
    Roupa a = new Roupa();
    Bolsa a = new Bolsa();

```

3. Dados três valores de tipo real (ponto flutuante) representando os lados de um triângulo:

- a. Escreva um método Java para determinar se os lados formam um triângulo, retornando verdadeiro ou falso, em cada um dos casos. **(1,5 pontos)**

```

if (x < y + z || y < x + z || z < x+y){
    System.out.println("Verdadeiro.Trata-se de um triangulo");
}else
    System.out.println("Falso. Não é um triângulo");

```

- b. Escreva um método Java para determinar o tipo de triângulo, retornando 1 para triângulo escaleno, 2 para triângulo isósceles e 3 para triângulo equilátero. Utilize o método definido no *item (a)* para definir se os lados formam triângulo antes de determinar o tipo deles. Retorne 0 se não for possível determinar o tipo de triângulo. **(1,5 pontos)**

```

if (x == y && x == z){
    System.out.println("1");
}else if(x == y || x == z){
    System.out.println("2");
}else
    System.out.println("3");

```