**Aluno:** Guilherme Silva da Cunha

**Turma:** 62168

**Lista 1 POO**

1) Sobrecarga é o nome dado quando o programador utiliza mais de um método, com nomes idênticos, e assinaturas diferentes. Ou um método que reage de forma diferente a cada situação.

2) Sobreposição é o ato de substituir um método existente através da criação de um método com a mesma assinatura.

3) Utilizando a palavra abstrata Super para acessar a superclasse.

4) Sim, é verdade.

5) Sim, uma classe abstrata pode conter métodos abstratos ou não abstratos.

6) Não podem serem privados, já que obrigatoriamente precisam ser sobrescritos.

7) Protected, Private, Public, Abstract, Default, Final, Static

8) Pode. Uma classe concreta pode ter métodos abstratos também.

9) Acrescenta "Final" na classe.

10) Acrescenta "Abstract" na classe.

11) É perfeitamente possível.

12) Diz respeito as responsabilidades de cada parte do produto. Quanto maior a coesão, melhor é feita a divisão das responsabilidades entre as classes

13) Quando um artefato ( atributo, método, tabela...) tem baixa dependência de outros artefatos.

**Lista 2 POO**

1)

a) É uma classe concreta.

b) É uma classe Abstrata.

c) É uma interface.

2)

a) Concreta.

b) 3 : Tanque, Status e Potencia.

c) 3 : LigarMotor, Abastecer, GetStatus. Não há assinaturas abstratas, e a classe é concreta.

d) Nenhuma. A classe motor não é abstrata.

e) Publico, dentro da classe e do pacote.

f) Private, somente a classe terá acesso.

g) Protected, dentro do pacote ou por subclasses.

h) Por que ela não tem Main.

i) A variável m1 recebe uma instância de Motor.

j) Tanque, Status, Potencia e os metodos : LigarMotor, Abastecer e getStatus.

k) O objeto exibirá a mensagem "Coloque gasolina",

l) Exibirá a mensagem " Ligado”

3)

a) Porque a classe Poligono tem um método abstrato.

b) Triangulo será uma subclasse Poligono, e poligono será a superclasse de Triangulo.

c) Poligono : qtdLados, area e metodos: setQtdLados, getQtdLados, calcularArea.

e Triangulo: qtdLados, area e metodos: setQtdLados, getQtdLados, calcularArea + base, altura, e sobrescreverá calcularArea.

d) Não. Por que ela não possui a palavra reservada abstract.

e) Podemos, pois ela é subclasse de Poligono.

f) Boas práticas, com a intenção de padronizar a forma de programar.

4)

a) Nenhuma, pois interfaces por padrão tem atributos public abstract. Duas abstratas : rolarDado e mostrarLado.

b) Não. Porque a interface obriga que a classe que a implementa sobrescreva seus métodos.

c) Não. Por que a classe mostrarLado é uma das classes abstract da interface que está sendo implementada.

d) Porque a obrigação desta classe é sobrescrever os métodos abstratos da interface, após eles terem sido sobrescritos, qualquer outra operação concreta poderá ser acrescentada que o programa irá ser compilado normalmente.

e) Não. Pois ao ter um método abstrato, a classe Dados6Lados obrigatoriamente deverá ser Abstrata.