

Curso/Turma:	Bacharelado em Engenharia de Software	NOTA
Disciplina:	Linguagem de Programação 1	
Tipo:		
Aluno(a): Jacieldo da Silva Gonçalves	Professor(a): Alcemir Santos Data: 10/10/2020	

Torne-se o líder que você quer ser!

Prova Mensal 1 REC

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

A orientação à objeto são paradigmas de programação e tem o objetivo de iteração das diversas unidades chamadas de objetos. Os objetos podem ser entendidos como algo do todo, como por exemplo, ao se falar de carro, o carro é um objeto, que pode-se definir o objeto pelos seus atributos (características), métodos (comportamento) e estado atual. E definir ou classificar um carro, você está fazendo uma classe do objeto carro. Na "classe carro", tem modelo; cor; peso; motorização; anda. A Orientação à objetos é um conjunto de objetos que interagem trocando mensagens através da aplicação de métodos à objetos.

2. Escolha dois dentre os conceitos pilares da orientação à objetos (abstração, encapsulamento, generalização e polimorfismo) e apresente a definição e qual seu papel na programação orientada à objetos. **(2 pontos)**

ABSTRAÇÃO - Do dicionário, "operação intelectual em que um objeto de reflexão é isolado de fatores que comumente lhe estão relacionados na realidade." É quando se abstrai do mundo real uma entidade para criar classes relativo ao problema que se deseja resolver. Os atributos e métodos das classes são abstraídos do mundo real. É através da abstração que se poderá definir as classes, subclasses e superclasses.

POLIMORFISMO - É o princípio a partir da qual uma subclasse é capaz de redefinir métodos herdados, que embora apresentem a mesma assinatura, comportem-se de maneira diferente para cada uma das classes derivadas. O polimorfismo serve para atributos e métodos serem utilizados em objetos distintos, mas com implementações lógicas diferentes.

3. Você está implementando um sistema de controle de matrículas de uma faculdade.
 - a. Defina uma classe Java para representar as disciplinas da faculdade e outra para as turmas de cada disciplina. **(1,5 pontos)**

```
public class Disciplinas {  
    String materia;
```

```
String professor;  
int horas;  
String curso;  
boolean emAndamento;  
void cursando() {  
    if (this.emAndamento == true) {  
        S.O.P("Disciplina em andamento");  
    } else {  
        S.O.P("As aulas da disciplina ainda não começarão");  
    }  
}
```

```
public class Turmas {  
    String curso;  
    String turma;  
    int qtdAlunos;  
    int semestre;  
    boolean status;  
  
    void aberta() {  
        this.status = true;  
    }  
    void fechada() {  
        this.status = false;  
    }  
    void vagas() {  
        if (this.status == true) {  
            S.O.P("Há vagas!");  
        } else {  
            S.O.P("Não há vagas!");  
        }  
    }  
}
```

- b. Considere-se escrevendo o método main. Escreva o código para instanciar disciplinas e turmas de definidas no item anterior. **(1,5 pontos)**

```
Public class Prova {  
    public static void main(String[] args) {  
  
        Disciplina1 d1 = new Disciplinas();  
        Turma1 t1 = new Turmas();  
    }  
}
```

4. Dados três valores, representando a quantidade de itens comprados, tipo de pagamento a ser utilizado e a quantidade de compras realizadas pelo cliente na loja:

- a. Escreva um método Java para determinar se o cliente é fiel à loja, retornando verdadeiro caso o número de compras for acima de 10 ou falso, em caso contrário. **(1,5 pontos)**

```
void fidelidade() {  
    if (this.compras > 10) {  
        this.fidelidade = true;  
    } else {  
        this.fidelidade = false;  
    }  
}
```

- b. Escreva um método Java para determinar o preço a ser cobrado pela compra. Utilize o método definido no *item (a)* para definir se o cliente é fiel, em caso afirmativo, adicione 5% de desconto. Adicione 5% de desconto ao valor já descontado por fidelidade, caso seja compra de mais que 3 itens ou aplique o desconto de 10% no valor inicial em caso de compra de mais que 5 itens. **(1,5 pontos)**

```
void desconto() {  
    if (this.fidelidade == true) {  
        this.preco = preco*0.95;  
    } else if(this.fidelidade == true && produtos > 3) {  
        this.preco = preco*0.95;  
        this.preco = preco*0.95;  
    } else if(produtos > 5) {  
        this.preco = preco*0.9;  
    }  
}
```

5. Considere um jogo de Campo Minado. Um jogo de tabuleiro. Cada zona do tabuleiro inicia coberta e deve, durante o jogo, ser revelada ou marcada como “contém bomba”. O objetivo do jogo é identificar a posição onde as bombas estão escondidas. Para descobrir esta informação, cada zona revelada que não continha uma bomba escondida apresenta um número indicando o perigo da zona. Cada zona tem no mínimo quatro (4) e no máximo oito (8) zonas adjacentes. O número que indica o perigo é a quantidade exata de bombas que estão distribuídas nas zonas adjacentes. Revelar uma zona que esconde uma bomba, significa que o jogador perdeu a partida e é o fim de jogo. A título de exemplo, figura abaixo apresenta um jogo perdido. Os números 1 no tabuleiro indicam a presença de uma bomba nas adjacência. Assim como os números 2 indicam duas bombas e os 3 e 4 indicam três e quatro bombas, respectivamente.



Você foi contratado para implementar o jogo utilizando orientação à objetos. Indique, ao menos, 5 Classes a serem utilizadas na construção do jogo de campo minado, incluindo, os atributos e métodos de cada uma delas. **(2 pontos)**