

Curso/Turma:	Bacharelado em Engenharia de Software		NOTA
Disciplina:	Engenharia de Software	Professor(a): Alcemir Santos	
Tipo:		Data: 18/09/2020	
Aluno(a):	Whalisson Kássio de Melo Frazão		

Torne-se o líder que você quer ser!

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

Classe: Um conjunto de atributos e métodos que vão definir as características e as ações dos objetos.

Objetos: São derivados das classes, e utilizam-se dos atributos e métodos da classe para se moldarem e praticarem ações predeterminadas no escopo da classe.

A Orientação à Objetos se dá pela abstração de um problema, criando classes e objetos relacionados a esse problema, que podem ou não interagir entre si, dependendo da finalidade do código. A interação irá se dar pela visibilidade dos métodos e atributos, que permitirá, ou não, a troca de mensagens entre eles.

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)**

```
public class ProdutosParaVenda(){
    String código;
    int quantidade;
    double precoDeCompra;
    double precoDeVenda;
}
public class ProdutosEmEstoque(){
    String código;
    int quantidade;
    double precoDeCompra;
    double precoDeVenda;
}
```

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

```
ProdutosParaVenda sapato = new ProdutosParaVenda("1214",45,65.4,125.4);
ProdutosEmEstoque chinelo = new ProdutosEmEstoque("1865",250,10.4,50.4);
```

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)**

```
public boolean eTriangulo(){
    if(ladoa > 0 && ladob > 0 && ladoc > 0){
        if(ladoa < ladob + ladoc && ladob < ladoa + ladoc && ladoc < ladoa + ladob){
            return true;
        }
    }
}
```

```
    }
  }
  return false;}

```

- 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)**

```
public int tipoTriangulo(){

    if(ladoa != ladob && ladob != ladoc && ladoc != ladoa){
        return 1;
    }
    if(ladoa == ladob || ladoa == ladoc){
        return 2;
    }

    if(ladoa == ladob && ladoa == ladoc){
        return 3;
    }

    return 0;
}

```

4.

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 adjacências. 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, 4 Classes a serem utilizadas na construção do jogo de campo minado, incluindo, os atributos e métodos de cada uma delas. **(2 pontos)**

```
public class Jogo{
    boolean ganhou;
    boolean jogoTerminado;
    int[] jogada;
    int rodada;
    public void jogar();
}
public class Tabuleiro{
    int[][] minas;
    String[][] tabuleiro;
    Public void limpaTabuleiro();
    public Boolean ganhou();

}
public class Minas implements Tabuleiro{
    public void sorteioMinas();
    public void startMinas();
    public void mostrarMinas();
}
```

Classe Random.

Classe Scanner.