



**UNIVERSIDADE FEDERAL DO CEARÁ – UFC SOBRAL**  
**TÉCNICAS DE PROGRAMAÇÃO – PROF. FISCHER**

**Lista 5: Programação Orientação a Objetos**

1. Crie uma classe Controle Remoto que pode controlar o volume e trocar os canais da televisão. O controle de volume permite:
  - aumentar ou diminuir a potência do volume de som em uma unidade de cada vez
  - aumentar e diminuir o número do canal em uma unidade trocar para um canal indicado
  - consultar o valor do volume de som e o canal selecionado
2. Criar a classe Pessoa com as seguintes características:
  - atributos: idade e dia, mês e ano de nascimento, nome da pessoa
  - métodos: o calculaIdade(), que recebe a data atual em dias, mês e anos e calcula e armazena no atributo idade a idade atual da pessoa o informaIdade(), que retorna o valor da idade o informaNome(), que retorna o nome da pessoa o ajustaDataDeNascimento(), que recebe dia, mês e ano de nascimento como parâmetros e preenche nos atributos correspondentes do objeto.
  - Criar dois objetos da classe Pessoa, um representando Albert Einstein (nascido em 14/3/1879) e o outro representando Isaac Newton (nascido em 4/1/1643)
  - Fazer uma classe principal que instancie os objetos, inicialize e mostre quais seriam as idades de Einstein e Newton caso estivessem vivos.
3. Crie uma classe Círculo para representar círculos. Esta classe deverá conter o seguinte atributo: raio e os métodos: área, circunferência, aumentar círculo.

OBS: no método aumentar círculo, receba um valor double e use-o como percentual para aumentar o raio do círculo.

Por fim, no método main(), crie um novo objeto da classe Círculo, peça para o usuário informar o raio e retorne a área e a circunferência. Peça também o percentual de aumento do raio e retorne a nova área e a nova circunferência.

OBS: Defina o pi como 3,14.

4. Crie uma classe Produto, essa classe deverá conter os seguintes atributos: nome do produto, preço de custo, preço de venda, margem de lucro e despesas da empresa. E o método: Cálculo da Margem de Lucros. OBS: Cálculo da Margem de Lucro:  $\text{Margem de Lucro} = \{[(PV - (PC + D)) / PV]\} \times 100$ . Com - PC= Preço de Custo, PV= Preço de Venda, D= Despesas. Por fim, no método main(), crie um novo objeto da classe Produto, peça para o usuário informar o nome, os preços de custo e de venda e exiba a margem de lucro em porcentagem.
5. A fim de representar empregados em uma firma, crie uma classe chamada Empregado. Esta classe deverá possuir as informações a seguir:
  - a. id
  - b. primeiro nome
  - c. sobrenome
  - d. salário mensal
  - e. um vetor de booleano que representa os dias trabalhados. Caso o dia foi trabalho é marcado na posição do vetor como true, caso não foi trabalho e marcado falso

Sua classe deve ter os seguintes construtores.

- Para receber o nome e sobrenome do empregado
- Para receber o nome e sobrenome do empregado além do id.

Sua classe deve poder calcular o salário do empregado, mediante a seguinte regra.

- O total de dias trabalhados for igual a 30 o salário será o salário base total com acréscimo de 1% do salário base
- O total de dias trabalhados for entre 25 a 27 dias para cada dia não trabalhado o empregado terá o desconto no seu salário 2% em relação ao seu salário base
- O total de dias trabalhados for entre 20 a 24 dias para cada dia não trabalhado o empregado terá o desconto no seu salário de 3% em relação ao seu salário base
- O total de dias trabalhados for menor que 20 o usuário receberá apenas 20% do seu salário base

6. Crie uma classe denominada Elevador para armazenar as informações de um elevador dentro de um prédio. A classe deve armazenar o andar atual (térreo = 0), total de andares no prédio (desconsiderando o térreo), capacidade do elevador e quantas pessoas estão presentes nele. A classe deve também disponibilizar os seguintes métodos:

Inicializa : que deve receber como parâmetros a capacidade do elevador e o total de andares no prédio (os elevadores sempre começam no térreo e vazio);

Entra : para acrescentar uma pessoa no elevador (só deve acrescentar se ainda houver espaço);

Sai : para remover uma pessoa do elevador (só deve remover se houver alguém dentro dele);

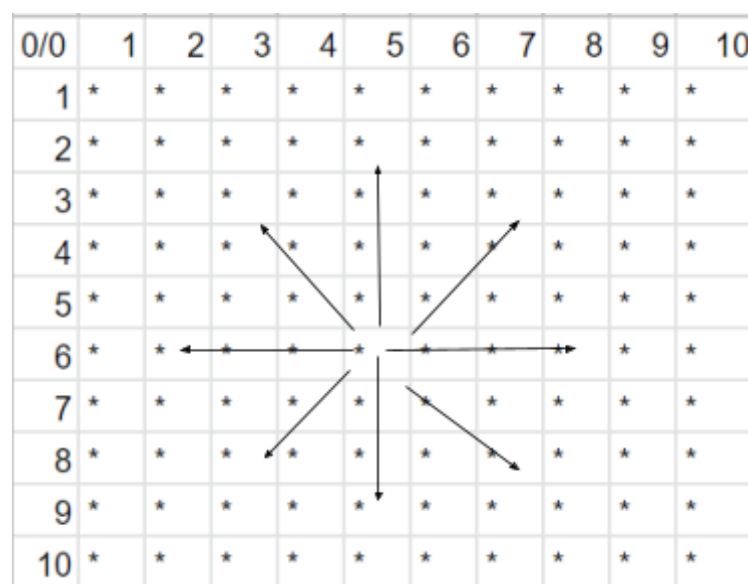
Sobe : para subir um andar (não deve subir se já estiver no último andar);

Desce : para descer um andar (não deve descer se já estiver no térreo);

Encapsular todos os atributos da classe (criar os métodos set e get).

7. Crie uma matriz para representar uma área em que um robô pode caminhar. Sendo que o robô poderá caminhar nas direções das setas indicadas abaixo. Cada direção deverá ser mapeada por um método na classe robô. O robô deverá caminhar apenas uma casa cada vez quando um método for chamado. O robô deverá ser colocado no meio da área sempre que o programa iniciar. O tamanho da área em que o robô caminha é quadrado, mas pode ter diferentes números de linhas e colunas

0/0	1	2	3	4	5	6	7	8	9	10
1	*	*	*	*	*	*	*	*	*	*
2	*	*	*	*	*	*	*	*	*	*
3	*	*	*	*	*	*	*	*	*	*
4	*	*	*	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*	*	*	*
8	*	*	*	*	*	*	*	*	*	*
9	*	*	*	*	*	*	*	*	*	*
10	*	*	*	*	*	*	*	*	*	*



- a) A classe deverá ter a contagem de quantas casas o robô caminhou nas direções: norte, sul, leste, oeste, nordeste, sudeste, sudoeste e noroeste.



III)

