

**GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**  
**DEPARTAMENTO DE COMPUTAÇÃO - UNIVERSIDADE ESTADUAL DE**  
**LONDRINA**

**LABORATÓRIO DE PROGRAMAÇÃO – PROF. BRUNO B. ZARPELÃO**

**LISTA DE EXERCÍCIOS – ATRIBUTOS E MÉTODOS**

**Instrução geral:** todos os atributos devem ser criados com modificador de acesso “private”. Para acessar/alterar os atributos, construa *getters* e *setters*

1. Construa um programa com os seguintes requisitos:
  - a. O programa deve ter uma classe “Carro” com os seguintes atributos: marca e modelo;
  - b. A classe Carro deve ter um método que recebe como parâmetro a velocidade atual do carro e retorna qual deve ser a marcha utilizada;
  - c. o programa deve conter uma classe *main* para que você teste a atribuição de valores aos atributos da classe Carro e a execução do método que retorna a marcha adequada a uma determinada velocidade;
2. Construa um programa com os seguintes requisitos:
  - a. O programa deve ter uma classe Funcionario.
  - b. A classe funcionário terá os seguintes atributos:
    - i. String nome;
    - ii. int salario;
    - iii. boolean ativo;
  - c. A classe funcionário terá os seguintes métodos:
    - i. Um construtor que permita o cadastramento inicial do nome e do salário do funcionário;
    - ii. Um método que receba como parâmetro um aumento em porcentagem para o salário do funcionário e aplique este aumento ao valor do atributo salário;
    - iii. Um método de demissão do funcionário que altere o valor do atributo “ativo” de verdadeiro para falso;
    - iv. Um método que mostre os valores dos atributos do funcionário na tela;
  - d. Faça uma classe *main* na qual você:
    - i. cadastre um funcionário e mostre os dados dele na tela;
    - ii. dê um aumento ao funcionário e mostre os dados dele na tela;
    - iii. demita o funcionário e mostre os dados dele na tela;
3. Construa um programa com os seguintes requisitos:
  - a. O programa deve ter uma classe Data;
  - b. A classe Data terá três atributos:
    - i. int dia;

- ii. `int mês;`
    - iii. `int ano;`
  - c. A classe `Data` deve ter dois métodos:
    - i. O primeiro método mostra a data na tela no formato `dd/mm/aaaa`;
    - ii. O segundo método mostra a data na tela como no exemplo: 12 de janeiro de 2012;
  - d. Faça uma classe `main` na qual você:
    - i. Cria uma data;
    - ii. Mostre esta data na tela no formato `dd/mm/aaaa`;
    - iii. Mostre esta data na tela no formato "11 de janeiro de 2012";
- 4. Construa um programa com os seguintes requisitos:
  - a. O programa deve ter uma classe `Pessoa`;
  - b. A classe `Pessoa` deve ter dois atributos:
    - i. `String nome`;
    - ii. `int idade`;
  - c. A classe `Pessoa` deve ter um método chamado `fazAniversario()`. Cada vez que o método for invocado, a idade da pessoa deve ser aumentada em um ano;
  - d. Faça uma classe `main` na qual:
    - i. Uma pessoa é cadastrada;
    - ii. O nome e a idade da pessoa são mostrados na tela;
    - iii. A pessoa faz cinco aniversários (utilize uma estrutura de repetição como `for` ou `while`);
    - iv. O nome e a idade da pessoa são mostrados na tela;
- 5. Construa um programa com os seguintes requisitos:
  - a. O programa deve ter uma classe `Casa`;
  - b. A classe `Casa` deve ter os seguintes atributos:
    - i. `String cor`;
    - ii. `boolean porta1`;
    - iii. `boolean porta2`;
    - iv. `boolean porta3`;
  - c. A classe `Casa` deve ter os seguintes métodos:
    - i. Um método `pintar(String novaCor)` que receba como argumento uma nova cor e mude o valor do atributo `cor`;
    - ii. Métodos para abrir e fechar as portas;
    - iii. Método para mostrar na tela a cor da casa e quantas portas estão abertas;
  - d. Faça uma classe `main` na qual:
    - i. Uma casa seja cadastrada;
    - ii. A cor e a quantidade de portas abertas sejam mostradas na tela;
    - iii. A cor da casa seja alterada;
    - iv. Portas sejam abertas ou fechadas (você escolhe);

v. A cor e a quantidade de portas abertas sejam mostradas na tela;

6. Construa um programa com os seguintes requisitos:

- a. O programa deve ter uma classe Funcionario.
- b. A classe Funcionário terá os seguintes atributos: salário/hora, quantidade de horas trabalhadas e numero de dependentes;
- c. A classe Funcionário terá um método que retorna o salário bruto do funcionário. Seguem informações para os calculos:
  - i.  $\text{Salário bruto} = \text{horas trabalhadas} * \text{salário hora} + (50 * \text{numero de dependentes})$ ;
- d. A classe Funcionário terá um método que retorna o salário liquido do funcionário. Seguem informações para os calculos:
  - i. Salário liquido é igual a salário bruto depois de descontos de INSS e IRPF;
  - ii. Desconto INSS:
    - Se  $\text{salário bruto} \leq 1000$ ,  $\text{desconto de INSS} = \text{salário bruto} * 8,5/100$
    - Se  $\text{salário bruto} > 1000$ ,  $\text{desconto de INSS} = \text{salário bruto} * 9/100$
  - iii. Desconto IRPF:
    - Se  $\text{salário bruto} > 500$  e  $\leq 1000$ ,  $\text{desconto de IR} = \text{salário bruto} * 5/100$
    - Se  $\text{salário bruto} > 1000$ ,  $\text{desconto de IR} = \text{salário bruto} * 7/100$
- e. Fazer classe principal que permita que:
  - i. O usuário insira o salário/hora, a quantidade de horas trabalhadas e o número de dependentes;
  - ii. O programa retorne na tela para o usuário o salário bruto e o salário líquido;

7. Construa um programa com os seguintes requisitos:

- a. O programa deve ter uma classe Operações;
- b. A classe Operações terá dois atributos:
  - i. `int valor1`;
  - ii. `int valor2`;
- c. A classe Operações deve ter um construtor que aceite como parâmetro dois valores inteiros e os atribua aos atributos `valor1` e `valor2`;
- d. A classe Operações deve ter métodos para realizar cada uma das operações básicas sobre estes valores: soma, multiplicação, divisão e subtração;
- e. Faça uma classe *main* na qual:
  - i. O usuário insere dois valores inteiros e a operação escolhida;
  - ii. O sistema retorna o valor da operação

8. Construa um programa com os seguintes requisitos:

- a. A ideia do programa é permitir que o usuário compare duas versões diferentes de um mesmo produto. Imagine que você está no supermercado e tem o achocolatado de 300 gramas por R\$ 4,00 e o achocolatado de 450 gramas por R\$ 5,50. Qual será mais barato?
  - b. O programa deve ter uma classe `CalcularPreco`;
  - c. A classe `CalcularPreco` terá os seguintes atributos:
    - i. `double preco1` (vai receber o preço do primeiro produto);
    - ii. `double preco2` (vai receber o preço do segundo produto);
    - iii. `double medida1` (vai receber a medida do primeiro produto, ex. 100 gramas);
    - iv. `double medida2` (vai receber a medida do segundo produto, ex. 200 gramas);
  - d. A classe `CalcularPreco` vai ter um método que retorna qual produto é mais barato: o produto 1 ou o produto 2;
  - e. Faça uma classe *main* na qual:
    - i. O usuário insere os valores e as medidas dos dois produtos;
    - ii. O sistema retorna qual produto é mais barato;
9. Construa um programa que calcule quanto um usuário irá pagar no valor de uma mensalidade escolar. O dia do vencimento da mensalidade é todo dia 10 de cada mês. Se o usuário pagar a mensalidade até o dia 05 do mês tem desconto de 10% na mensalidade. Caso pague até o dia 09 tem 5% de desconto na mensalidade. Se pagar no dia 10, não há desconto. Se o dia de pagamento ultrapassar a data de vencimento, terá um acréscimo de 5% no valor da mensalidade. Você terá que definir quais são as classes e atributos para este exercício.