

Sexta Lista de Exercícios

Classes e Objetos

Norton Trevisan Roman

30 de março de 2020

1. O que fazem os métodos a (da classe A) e b (da classe B)? Não precisa entrar em detalhes.

```
class A {
    double a(int meses, double taxa) {
        return Math.pow((taxa + 100) / 100, meses) - 1;
    }
}

class B {
    final double TAXA = 1.2;

    void b(double valorEmprestado, int meses) {
        A a = new A();
        double valorDaDívida = valorEmprestado + (a.a(meses, TAXA) *
                                                    valorEmprestado);
        System.out.println("Dívida de "+valorDaDívida+" real(is)," +
                           "calculada com taxa de "+TAXA+"% ao mês.");
    }
}
```

2. Escreva uma classe **Olah** com um único método `cumprimenta` que, a cada chamada, cumprimenta o usuário de uma entre 3 maneiras diferentes. Dica: use um atributo para, dependendo de seu valor, escolher qual das maneiras será usada; depois de imprimir a mensagem, altere o valor do atributo.
3. Escreva uma classe **Simples** contendo um método que recebe um arranjo de inteiros como parâmetro e que inicializa todos os elementos do arranjo com um valor, também dado como parâmetro. O método deve devolver o tamanho do arranjo. A assinatura do método deve ser a seguinte:

```
int inicializaArranjo (int[] a, int v);
```

Escreva agora um método que recebe um arranjo de inteiros como parâmetro e imprime o seu conteúdo:

```
void imprimeArranjo (int[] a) ;
```

Crie agora um método que, dado um inteiro, verifica se ele está presente no arranjo.

```
boolean estaNoArranjo (int[] a, int v) ;
```

Finalmente, escreva uma classe que cria um arranjo, cria um objeto `Simples` e chama seus três métodos.

4. Construa a classe `Inteiro` que representa um número inteiro. Essa classe deve ter os seguintes atributos e métodos:

- Atributos:
 - `int valor` – Valor do inteiro representado.
- Métodos para interação com o usuário da classe:
 - `void carregaValor(int v)` – Muda o valor representado por este objeto. O novo valor deve ser `v`.
 - `int devolveValor()` – Devolve o valor representado por este objeto.
 - `int devolveValorAbsoluto()` – Devolve o valor absoluto do valor representado por este objeto.
 - `void imprime()` – Imprime algo que representa este objeto. Sugestão: imprima o seu valor.
 - `int soma(int v)` – Soma `v` ao valor deste objeto (`valor + v`). Este objeto passa a representar o novo valor, que também deve ser devolvido pelo método.
 - `int subtrai(int v)` – Subtrai `v` do valor deste objeto (`valor - v`). Este objeto passa a representar o novo valor, que também deve ser devolvido pelo método.
 - `int multiplicaPor(int v)` – Multiplica o valor deste objeto por `v` (`valor * v`). Este objeto passa a representar o novo valor, que também deve ser devolvido pelo método.
 - `int dividePor(int divisor)` – Verifica se `divisor` é diferente de zero. Se não, imprime uma mensagem de erro e não faz nada (devolve o valor inalterado). Se for, divide o valor deste objeto por `v` (`valor / divisor`, divisão inteira). Este objeto passa a representar o novo valor, que também deve ser devolvido pelo método.

5. Crie uma classe `TestaInteiro` que verifica o funcionamento correto da classe `Inteiro` (apresentada acima) em diversas situações (criando objetos `Inteiro` e testando-os). Lembre-se de verificar casos como a divisão por diversos valores.

6. Crie a seguinte classe que represente figuras esféricas:

- Atributos: `Raio`
- Métodos:
 - `area()` – calcula a área da superfície da esfera
 - `volume()` – calcula o volume da esfera

7. Defina uma classe `Complexo` e escreva métodos para efetuar a soma, subtração, multiplicação e divisão de números complexos. Escreva métodos também que, dado um complexo, retornem seu conjugado e seu módulo.

8. Você foi contratado por uma agência de viagens para escrever uma classe em Java para calcular a conversão de reais para dólar de acordo com a taxa de compra e a taxa de venda. Para isso, escreva uma classe `ConversorMonetário` que inclua os seguintes métodos:

- `defineTaxaCompra()`
- `defineTaxaVenda()`
- `imprimeTaxas()`, que imprime o valor das 2 taxas de conversão;
- `vendeDólar()`, que recebe uma quantia em dólares e retorna o valor correspondente em reais
- `compraDólar()`, que recebe uma quantia em dólares e retorna o valor correspondente em reais

Em seguida, escreva uma classe `TestaConversorMonetário` que define diferentes taxas de compra e venda de dólares e, para cada taxa de conversão, realiza operações de compra e venda.

9. Crie uma classe contendo um método que, dado um ponto determinado pelas suas coordenadas x e y , reais, imprime em qual quadrante este ponto está localizado. O primeiro quadrante corresponde aos pontos que possuem x e y positivos, o segundo quadrante a x positivo e y negativo e assim por diante.
10. Escreva uma classe `Rendimentos` que contenha os seguintes métodos a fim de contabilizar o total de rendimentos de uma certa pessoa em um certo ano:

- `rendimentosDePessoaFísica(double);`
- `rendimentosDePessoaJurídica(double);`
- `rendimentosDeAplicaçõesFinanceiras(double);`
- `rendimentosNãoTributáveis(double);`
- `double totalDeRendimentosTributáveis();`

Em seguida, escreva uma classe `TabelaDeAliquotas` que possui:

- um método `aliquota()` que recebe como parâmetro o total de rendimentos tributáveis de uma pessoa e devolve um número entre 0 e 1.0 correspondente à alíquota de imposto que a pessoa deverá pagar e
- um método `valorADeduzir()` que recebe como parâmetro o total de rendimentos tributáveis de uma pessoa e devolve o valor a deduzir no cálculo do imposto.

Para escrever esta classe, utilize a tabela anual do IRPF 2017 abaixo:

Rendimentos Tributáveis	Alíquota	Parcela a deduzir
Até R\$ 22.847,76	—	—
De R\$ 22.847,77 até 33.919,80	0,075	R\$ 1.713,58
De R\$ 33.919,81 até 45.012,60	0,15	R\$ 4.257,57
De R\$ 45.012,61 até 55.976,16	0,225	R\$ 7.633,51
Acima de R\$ 55.976,16	0,275	R\$ 10.432,32

Agora escreva uma classe `CalculadoraDeImposto` que possui um único método que recebe como parâmetro o valor dos rendimentos tributáveis de uma pessoa e devolve o valor do imposto a ser pago.

11. Reescreva a classe `CalculadoraDeImposto`, contendo um método que não recebe nenhum parâmetro e devolve o imposto a ser pago. A classe `CalculadoraDeImposto` deve possuir em seus atributos as classes `Rendimentos` e `TabelaDeAlíquotas` e receber como parâmetros em seu construtor todos os valores necessários para inicializar objetos destas classes.

12. Escreva uma classe **MediaAlunoDisciplina** que tenha os atributos número do aluno, nome da disciplina, peso de trabalhos, peso de provas, média de trabalhos, média de provas, média ponderada, com as seguintes especificações:

- o número do aluno, o nome da disciplina devem ser fornecidos no momento da criação do objeto;
- os pesos de provas e trabalhos são variáveis (atributos) da classe. A soma desses atributos sempre deverá resultar no valor 10;
- o nome da disciplina também é uma variável da classe;

Crie os seguintes métodos (faça as considerações necessárias sobre tipos de atributos, métodos e especificadores de acesso):

- **calculaMediaPonderada()** – calcula a média do aluno, multiplicando as médias de trabalhos e provas pelos seus respectivos pesos, e altera o atributo que armazena a média ponderada. Este atributo nunca poderá ser maior que 10;
- **imprimeDados()** – imprime todos os atributos;

13. Defina uma classe **IntervaloDeTempo** cujos objetos representam um intervalo de tempo em número de horas, minutos e segundos. O construtor de objetos dessa classe deve receber como argumento um número inteiro positivo, representando o número de segundos decorridos desde o instante inicial 00:00:00 horas e retornar um objeto da classe **IntervaloDeTempo** correspondente.

Por exemplo, a expressão `new IntervaloDeTempo(3500)` deve retornar um objeto que represente 0 horas, 58 minutos e 20 segundos. Crie também uma classe para testar a classe definida.

14. Defina uma classe **Pessoa** cujos objetos representam uma pessoa contendo as seguintes informações: nome, sexo, número do documento de identidade e data de nascimento da pessoa (dia, mês e ano). Defina um construtor de objetos dessa classe e também um método **idade()** para retornar a idade da pessoa (um número inteiro de anos) dado como parâmetro a data atual. Caso a data passada como argumento seja anterior à data de nascimento da pessoa o método deve retornar -1.