

# Programação Orientada a Objetos I

## Exercícios: exceções

### 1 Exercícios

1. Dado o trecho de código abaixo:

```
int[] vetor = new int[] { 2, 4, 6, 8, 10, 12 };
for (int i = 0; i <= 12; i++) {
    System.out.println(vetor[i]);
}
```

Implemente um programa em Java que execute este trecho. Execute o programa e veja a exceção produzida por acesso a posição fora dos limites do vetor. Trate esta exceção, imprimindo na tela uma mensagem dizendo que o vetor acabou.

2. Dado o trecho de código abaixo:

```
public static void metodo01() {
    Class.forName("ClasseQueNaoExiste");
}

public static void metodo02() {
    java.io.File.createTempFile("pre", "suf");
}

public static void metodo03() {
    Integer.class.newInstance();
}

public static void main(String[] args) {
    metodo01();
    metodo02();
    metodo03();
}
```

Uma classe com estes quatro métodos não compila. Quais passos são necessários para fazê-la compilar? Altere o código para que a classe compile sem erros.

3. Modifique a classe que representa a conta-corrente do exercício 1 da lista de exercícios 3, lançando uma exceção (criada por você) caso o usuário tente depositar ou sacar um valor negativo. Adicione uma outra exceção (também criada por você) caso o usuário tente sacar um valor maior do que o saldo da conta dele. Trate as exceções no seu programa principal, exibindo mensagens de erro adequadas.