



Universidade Federal do Ceará  
Campus de Sobral - Engenharia da Computação  
Técnicas de Programação – TD04  
Prof. Iális Cavalcante ([ialis@sobral.ufc.br](mailto:ialis@sobral.ufc.br))

**Entrega: 12/12/2015**

**Questão 1**

**Pacote: td04.questao1**

Crie sua própria exceção seguindo o modelo abaixo, em seguida construa uma função para calcular a raiz de um determinado número que gere uma exceção (Cláusula throws), caso o número seja negativo, que será tratada no método principal. Ao final, independentemente do que acontecer, deve imprimir a mensagem “fim”.

```
public class NovaExcecao extends Exception{  
    public NovaExcecao() {  
        super ("Houve um novo erro!");  
    }  
    public NovaExcecao (String msg){  
        super (msg);  
    }  
}
```

**Questão 2**

**Pacote: td04.questao02**

**Capturando exceções com superclasses**

Utilize herança para criar uma superclasse de exceção (chamada **ExceptionA**) e subclasses da exceção **ExceptionB** e **ExceptionC**, em que **ExceptionB** herda de **ExceptionA** e **ExceptionC** herda de **ExceptionB**. Escreva um programa para demonstrar que o bloco catch para o tipo **ExceptionA** captura exceções de tipos **ExceptionB** e **ExceptionC**.

### Questão 3

**Pacote: td04.questao03**

Analise o trecho de código abaixo e identifique as possíveis exceções, em seguida implemente um bloco try-catch que seja capaz de capturar todas as exceções e imprimir uma mensagem diferente para cada uma.

```
int numero = Integer.parseInt( entrada.readLine() );
int matriz[][] = {{0, 1, 2}, {3, 4}};
matriz[1][2] = numero;
int x = 2;
x /= numero;
System.out.println("O último elemento da matriz é: "+numero);
```