



Universidade Federal da Fronteira Sul
Ciência da Computação
Programação I - Prof. Fernando Bevilacqua

Exercícios sobre pacotes e exceções

1) Qual a saída gerada pelo código abaixo? Justifique sua resposta.

<pre>class EGeral extends Exception {} class EMovimento extends EGeral {} class ERecurso extends EGeral {} class ECaminho extends EMovimento {}</pre>	<pre>void analisa(int i) throws EGeral { if(i < 0) { throw new ECaminho(); } if(i == 0) { throw new ERecurso(); } if(i == 9) { throw new EGeral(); } }</pre>
---	---

```
try {
    int j = -2;
    try {
        analisa(j);

        } catch(ERecurso a) { System.out.println("Recurso!"); }
        } catch(EMove a) { System.out.println("Move!"); }
        } catch(ECaminho a) { System.out.println("Caminho!"); }
        } catch(EGeral a) { System.out.println("Geral!"); }
        } finally { j = 9; }
    analisa(j);
} catch(ECaminho a) { System.out.println("Caminho!"); }
} catch(EMove a) { System.out.println("Move!"); }
} catch(ERecurso a) { System.out.println("Recurso!"); }
} catch(Exception a) { System.out.println("Excecao!"); }
```

2) Dado o código abaixo:

```
class ErroInterpretador extends Exception {}
class ErroSintaxe extends ErroInterpretador {}
class ErroExecucao extends ErroInterpretador {}
class TokenDesconhecido extends ErroSintaxe {}
class ComandoMalFormado extends ErroSintaxe {}
class ValorInvalido extends ErroExecucao {}
```

```
try {
    Gerenciador g = new Gerenciador();
    String s;
    // s é instanciado com alguma informação
    g.analisa(s);
```

```
} catch(ErroSintaxe a) { System.out.println("Erro Sintaxe!"); }
} catch(TokenDesconhecido b) { System.out.println("Token Desconhecido!"); }
} catch(ComandoMalFormado c) { System.out.println("Comando Mal Formado!"); }
} catch(ValorInvalido d) { System.out.println("Valor Invalido!"); }
} catch(ErroExecucao e) { System.out.println("Erro Execucao!"); }
```

a) Implemente a classe `Gerenciador`, com o método `analisa()`. Se a string passada como parâmetro possuir tamanho igual a zero, o método levanta a exceção `TokenDesconhecido`; se a string possuir tamanho igual a 5, o método levanta a exceção `ComandoMalFormado`. Para todos os outros casos, o método não levanta exceção alguma.

b) Suponha que o método `analisa()` levantou a exceção `ComandoMalFormado` durante a execução do código acima, qual será a saída do programa? Justifique sua resposta.



Universidade Federal da Fronteira Sul
Ciência da Computação
Programação I - Prof. Fernando Bevilacqua

3) Dadas as classes:

```
class NetworkException extends Exception {}  
class WirelessException extends NetworkException {}  
class EthernetException extends NetworkException {}  
class LostSignalException extends WirelessException {}  
class ChecksumException extends WirelessException {}
```

a) Escreva um método chamado `makeConnection()` que receba um inteiro como parâmetro e retorne um inteiro que equivale ao dobro do valor do parâmetro. Se o valor do parâmetro for 0 o método deve levantar uma `ChecksumException`, se for 1 deve levantar uma `LostSignalException`, se for 2 deve levantar uma `EthernetException`, se for 3 deve levantar uma `WirelessException`, se for 4 deve levantar uma `NetworkException`, caso contrário não levanta exceções.

b) Com base no método criado na questão a), o que aparecerá na saída padrão se o código abaixo for executado com `value` tendo valor 1? Justifique.

c) Com base no método criado na questão a), o que aparecerá na saída padrão se o código abaixo for executado com `value` tendo valor 2? Justifique.

d) Com base no método criado na questão a), o que aparecerá na saída padrão se o código abaixo for executado com `value` tendo valor 90? Justifique.

```
try {  
    makeConnection(value);  
  
} catch (ChecksumException x1) {  
    System.out.println("exception ChecksumException ");  
  
} catch (WirelessException x2) {  
    System.out.println("exception WirelessException ");  
  
} catch (EthernetException x3) {  
    System.out.println("exception EthernetException ");  
  
} catch (NetworkException x4) {  
    System.out.println("exception NetworkException ");  
  
} catch (LostSignalException x5) {  
    System.out.println("exception LostSignalException ");  
  
} finally {  
    System.out.println("closing connection");  
    return;  
    System.out.println("flushing log...");  
}
```



Universidade Federal da Fronteira Sul
Ciência da Computação
Programação I - Prof. Fernando Bevilacqua

4) Considerando as classes:

```
interface Rastreador {  
}  
class Animal {  
    private int idade, tamanho;  
    public void respirar() {}  
    public void andar() {}  
}  
class Mamifero extends Animal {  
    public void mamar() {}  
}  
class Morcego extends Mamifero implements Rastreador {  
    private float sonar;  
    public void voar() {}  
}
```

Assumindo que a interface `Rastreador` está no pacote `br.edu.uffs.cc.pr.habilidades`, que a classe `Animal` está no pacote `br.edu.uffs.cc.pr.entidades`, que as classes `Mamifero` e `Morcego` estão no pacote `br.edu.uffs.cc.pr.racas`, e que a classe que possui o método `main()` está no pacote `br.edu.uffs.cc.pr`. Além disso considere que o código do seu projeto esteja na pasta `Proj`. Responda:

- Como seria a estrutura de pastas e arquivos dentro desse diretório `Proj` para que o sistema de pacotes funcionasse corretamente?
- Qual instrução deve ser adicionada no começo do arquivo de cada classe para que ela faça parte do pacote? Mostre o comando usado para cada uma das classes.
- Faça o código completo da classe com o método `main()` para que ela instancie cada uma das classes citadas.