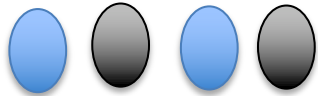




CIÊNCIAS DA COMPUTAÇÃO

FACULDADE DE CIÊNCIAS | UNIVERSIDADE AGOSTINHO NETO



FUNDAMENTOS DE PROGRAMAÇÃO

Docente:

✓ Lufialuiso Sampaio Velho, MSc.

Monitor

✓ João Pedro



Cap. IX

- Exceções

CAP. IX – Exceções (Exceptions)

Motivação

Quando criamos um programa, muita das vezes deparámo-nos com situações que não esperamos ou aquelas que ocorrem por esquecimento. Refiro-me propriamente das seguintes situações: utilização de uma variável não declarada, divisão de um número por zero, leitura de um tipo de dado inadequado (`int` - `String`), acesso a uma posição do vector fora dos limites, etc.

CAP. IX – Exceções (Exceptions)

Motivação (cont.)

Cada uma das situações acima e outras não referenciadas geram erros; ou seja mensagens com **letras vermelhas** na consola do IDE (**compilador**) da qual chama-mos de **exceções**.

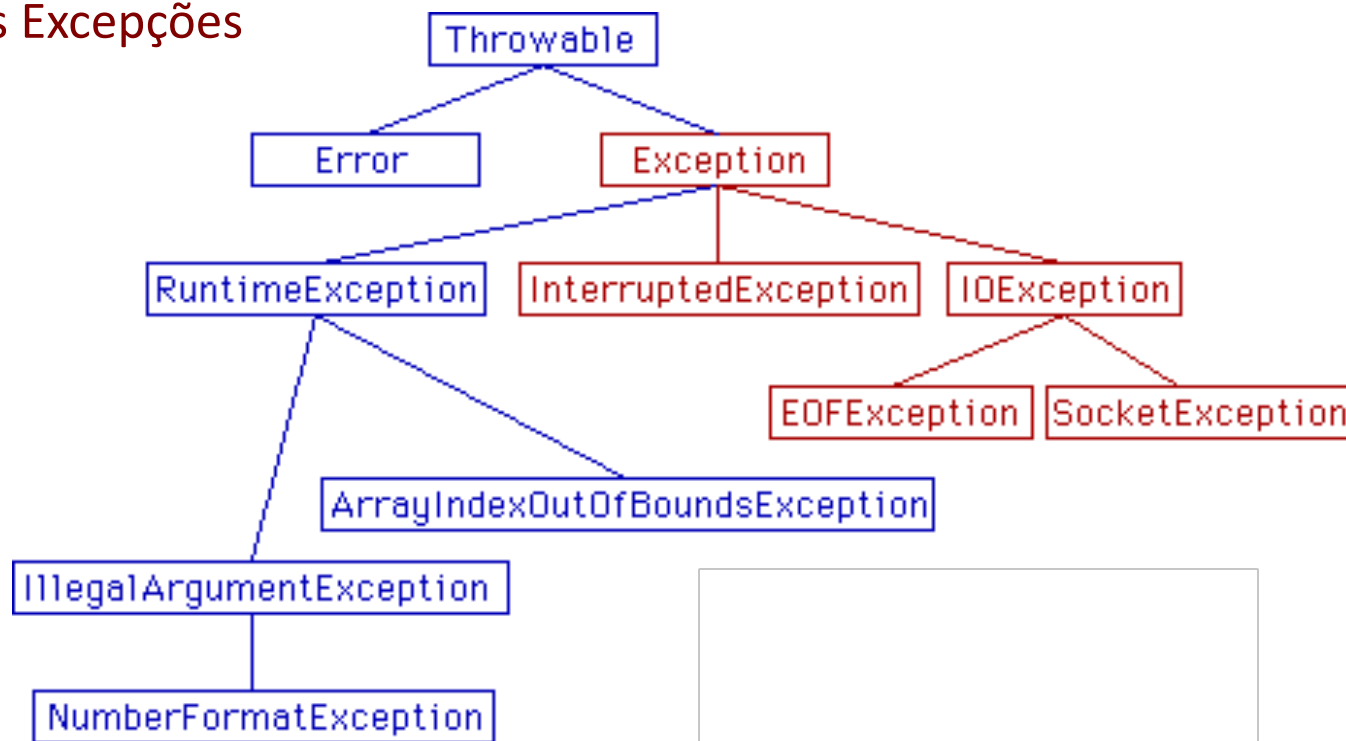
Definição

As exceções representam problemas que ocorrem durante a execução do programa, desviando assim o funcionamento normal do mesmo.

CAP. IX – Exceções (Exceptions)

Todas exceções são **objectos**. Isto quer dizer que existem classes apropriadas para cada **excepção** gerada e podem ser do tipo **Exception** ou do tipo **Error**; todas elas são subclasses da classe **Throwable**.

Hierarquia das Exceções



CAP. IX – Exceções (Exceptions)

Error: representam exceções de difícil ou impossível recuperação.

Ex: insuficiência de memória, erros de sintaxe, etc.

Quando estes ocorrem, causam imediatamente o encerramento do programa.

Exception: representam situações inesperadas mas que podem ser tratadas. Para a classe **Exception** existem as **RuntimeException** e as **não RuntimeException**.

CAP. IX – Exceções (Exceptions)

As `RuntimeException` ocorrem no momento da execução. São exceções do tipo `RuntimeException` as seguintes:

- `IndexOutOfBoundsException`,
- `IllegalArgumentException`,
- `InputMismatchException`,
- `ArrayIndexOutOfBoundsException`, etc.

CAP. IX – Exceções (Exceptions)

As outras exceções não consideradas como Runtime, podem ser:

- IOException,
- ClassNotFoundException,
- FileNotFoundException,
- UnknownHostException, etc.

CAP. IX – Exceções (Exceptions)

As exceções estudadas podem ser consideradas como **verificadas / não verificadas**.

Exceções verificadas: representam exceções detectadas no momento da compilação do programa; isto é estão fora do controlo do programa.

P.e: tentativa de acesso a um arquivo.

Excepção não verificada: representam exceções não detectadas na compilação do programa mas sim no momento de execução (Runtime).

Estas são normalmente causadas por uma insuficiência do programa.

P.e: leitura de um caractere enquanto era esperado um número.

CAP. IX – Exceções (Exceptions)

Tratamento de Exceções

Quando uma Exception ocorre, é necessário que seja tratada por mecanismos de tratamento de Exceções.

Estes mecanismos são propriamente os blocos **try** e **catch**.

Nota: não te esqueça que o tratamento de exceções é apenas para a classe Exception e suas subclasses e não para a Error.

CAP. IX – Exceções (Exceptions)

O **bloco try** do Java é usado para colocar o código que pode lançar uma exceção.

- Deve ser usado dentro de um método

Cada **bloco try** deve ser seguido por um **bloco catch** ou **finally**

- Ambos os tipos de blocos podem estar presentes. É obrigatório ter pelo menos um (**bloco catch** ou **finally**).

Um **bloco catch** está associado a um determinado tipo de exceção.

- Especifica o tratamento de exceção para esse tipo de exceção;
- Podem ser usados vários **blocos catch** para um único **bloco try**;
- Uma instrução **catch** não pode capturar uma exceção lançada por outra instrução **try**

CAP. IX – Exceções (Exceptions)

Sintaxe:

```
try{
```

```
// Instruções que podem gerar exceções
```

```
} catch (objecto da classe de exceção){
```

```
// instruções de captura e tratamento de exceção
```

```
}
```

Bloco finally

- usado para executar códigos importantes como fechar conexão, fluxos, etc.
- Este bloco é sempre executado independentemente se uma exceção tenha ocorrido ou não

Até a próxima Aula

