



C++

Tratamento de Exceções



Softblue
cursos online

Anomalias

- Situações inesperadas e/ou não imaginadas que podem ocorrer
- Exemplos clássicos de anomalias
 - Entrada de dados inválida
 - Falta de espaço em disco
 - Falta de permissão de acesso
 - Falha em hardware
- Quanto mais anomalias forem detectadas melhor será a solução via código

Exceptions

- Recurso disponibilizado pela linguagem para tratamento de anomalias no sistema
- A implementação de exceções varia de desenvolvedor para desenvolvedor, mas os recursos não
- Dependente do compilador, a maioria dos mais recentes já apresentam suporte



Anomalia sem tratamento



exemplo

```
double divisao (double a, double b)
{
    return a/b;
}
```

- Erro lançado se $b = 0$
- Programa encerra a sua execução
- Usuário e desenvolvedor ficam sem informações e/ou tratamentos do que ocorreu

Anomalia tratamento simples



exemplo

```
double divisao (double a, double b)
{
    if(b==0)
        return -1;
    else
        return a/b;
}
```

- Torna possível o tratamento pelo desenvolvedor
- Fica a cargo do desenvolvedor tratar
- Não é 100% eficaz

Diretivas



- Invoca uma função com exception: **try**
 - Indica que o bloco de código poderá lançar uma anomalia para ser tratada
- Trata a exception: **catch**
 - Bloco de ações a serem realizadas caso uma anomalia ocorra
- Lança uma exception: **throw**
 - Interrompe o código retornando ao final do bloco de tratamento, continuando a execução após o mesmo

Anomalia tratada com Exception



exemplo

```
double divisao (double a, double b) throw (int)
{
    if(b == 0)
        throw(-1);
    else
        return a/b;
}
```

- Exception lançada caso uma anomalia seja detectada
- Desenvolvedor é informado

Chamando função com exception



exemplo

```
try
{
    int var = divisao(15/0);
}
catch(int e)
{
    // Ações para tratamento da anomalia
}
```

- Encerramento do programa é opcional
- Possibilidade de logar o erro
- Informações ao usuário

Exceptions aninhadas



- Tipo da exception define o tratamento

exemplo

```
int teste(double a) throw(int, float, double)
{
    if(a == 0) {
        throw(int(1));
    }
    else if(a == 1) {
        throw(float(1.0));
    }
    else if(a == 2) {
        throw(double(1.0));
    }
}
```

exemplo

```
try {
    int var = divisao(15/0);
}
catch(int e) {
    // Ações para tratamento da anomalia
}
catch(float e) {
    // Ações para tratamento da anomalia
}
catch(double e) {
    // Ações para tratamento da anomalia
}
```

Exceptions desconhecida

- Tratando uma exception genérica (...)

exemplo

```
int teste(double a) throw(int, float, double)
{
    if(a == 0) {
        throw(int(1));
    }
    else if(a == 1) {
        throw(float(1.0));
    }
    else if(a == 2) {
        throw(double(1.0));
    }
}
```

exemplo

```
try {
    int var = divisao(15/0);
}
catch(int e) {
    // Ações para tratamento da anomalia
}
catch(...) {
    // Ações para tratamento da anomalia
}
```

Criando uma exception própria

- Implementar uma classe exception

exemplo .h

```
class minhaException {
public:
    minhaException();
    void message();
}
```

exemplo .cpp

```
#include ".h"

void minhaException::message() {
    // Ações para tratamento da anomalia
}
```

exemplo aplicação

```
try {
    int var = divisao(15/0);
}
catch(minhaException e) {
    // Ações para tratamento da anomalia
}
```

Aulas práticas e manuais on-line



Assista agora as aulas práticas.

[Clique aqui](#) para visualizar as aulas práticas disponíveis
