

DISCIPLINA: PROJETO DE SISTEMAS APLICADO AS MELHORES PRÁTICAS EM QUALIDADE DE SOFTWARE E GOVERNANÇA DE TI

**AULA:
14– TESTE DE SOFTWARE -EXERCÍCIOS**

**PROFESSOR:
RENATO JARDIM PARDUCCI**

PROFRENATO.PARDUCCI@FIAP.COM.BR

[Renato Parducci - YouTube](#)

EXERCÍCIOS DE FIXAÇÃO RESOLVIDOS

1. EXERCÍCIO RESOLVIDO

Crie um caso de teste unitário para o algoritmo a seguir (Método XPTO01).

Classe de Objetos MinMax1

inteiro n1, n2;

Método Comp_Numeros()

Início_Método

Exibir_mensagem "Entre com o primeiro inteiro positivo: "

n1 = Digitar_Valor

Exibir_mensagem "Entre com o segundo inteiro positivo: "

n2 = Digitar_Valor

Se n1 > 0 ou n2 > 0

Se (n1 > n2)

Mens = Concatenar("O menor numero é: ", n2)

Exibir_mensagem Mens;

Senão

Mens = Concatenar("O menor numero é: ", n1)

Exibir_mensagem Mens

Fim_Se

Senão Exibir_mensagem "Informe somente números positivos não nulos"

Fim_Se

Fim_Método

Fim_Classe

Use o método de elaboração de testes que permite checar cada caminho lógico da aplicação, pelo menos 1 vez!

EXERCÍCIOS DE FIXAÇÃO PROPOSTOS

2. DESAFIO

Crie um teste de Caixa Branca para o seguinte programa de aplicação:

```
using namespace std;
int r=0; //variável global começa em 0

int main (void){
    r++;

    { // início do bloco 1
        int x; // x é definida no bloco 1
        ...; // x e r podem ser usadas aqui
        { // início do bloco 2
            int y; // y é definida no bloco 2
            ...; // x, r e y podem ser usadas aqui
        } // y é automaticamente destruída no final do bloco 2 mas x e r não
        ...; // x ainda pode ser usada daqui em diante
    } // x é automaticamente destruída no final do bloco 1 mas r não
```

3. DESAFIO

Você está desenvolvendo um projeto que utilizará sensores de luz e temperatura, os quais serão colocados em um ambiente qualquer para capturar sinais desse local. Os sensores estarão conectados a um dispositivo Arduíno cada um e esses dispositivos transmitirão dados para um computador ao qual serão conectados por uma rede de dados qualquer.

Os dados exportados por esses dispositivos a cada minuto, serão guardados em arquivos que ficam à disposição para consumo de informações em um sítio na Internet.

Um software de retaguarda irá acessar os arquivos na nuvem e fazer uma associação de temperatura e luminosidade do ambiente.

Se a luminosidade superar 10.000 lumens e o calor exceder 60 graus Célsius, o software emitirá uma mensagem de alerta de incêndio.

Defina, aplicando um método de concepção de teste:

- Um caso de teste de nível unitário, do tipo funcional, com técnica de teste de caixa preta para a parte relacionada à associação da temperatura com a luminosidade, do sistema de gerenciamento de risco reportado no slide anterior. Imagine que você construirá um componente que lerá os dados de luz e calor de minuto em minuto, sendo que esse programa deve gerar um alarme em caso de detecção de possível incêndio

4. DESAFIO

Crie um caso de teste unitário com regra de Condição e Equivalência

Classe de Objetos MinMax1

inteiro n1, n2;

Método Comp_Numeros()

Início_Método

Exibir_mensagem "Entre com o primeiro inteiro positivo: "

n1 = Digitar_Valor

Exibir_mensagem "Entre com o segundo inteiro positivo: "

n2 = Digitar_Valor

Se n1 > 0 ou n2 > 0

Se (n1 > n2)

Mens = Concatenar("O menor numero é: ", n2)

Exibir_mensagem Mens;

Senão

Mens = Concatenar("O menor numero é: ", n1)

Exibir_mensagem Mens

Fim_Se

Senão Exibir_mensagem "Informe somente números positivos não nulos"

Fim_Se

Fim_Método

Fim_Classe

5. DESAFIO

Crie um teste de Caixa Branca para o seguinte programa de aplicação – escolha uma das técnicas de desenvolvimento de casos de teste que você aprendeu:

```
using namespace std;
int r=0; //variável global começa em 0

int main (void){
r++;

{ // início do bloco 1
  int x; // x é definida no bloco 1
  ...; // x e r podem ser usadas aqui
  { // início do bloco 2
    int y; // y é definida no bloco 2
    ...; // x, r e y podem ser usadas aqui
  } // y é automaticamente destruída no final do bloco 2 mas x e r não
  ...; // x ainda pode ser usada daqui em diante
} // x é automaticamente destruída no final do bloco 1 mas r não
```

**SOLUÇÃO DOS EXERCÍCIOS
DE FIXAÇÃO RESOLVIDOS**

SOLUÇÃO

Classe de Objetos MinMax1

inteiro n1, n2;

Método Comp_Numeros()

Início_Método

Exibir_mensagem "Entre com o primeiro inteiro positivo: "

n1 = Digitar_Valor

Exibir_mensagem "Entre com o segundo inteiro positivo: "

n2 = Digitar_Valor

Se n1 > 0 ou n2 > 0

Se (n1 > n2)

Mens = Concatenar("O menor numero é: ", n2)

Exibir_mensagem Mens;

Senão

Mens = Concatenar("O menor numero é: ", n1)

Exibir_mensagem Mens

Fim_Se

Senão Exibir_mensagem "Informe somente números positivos não nulos"

Fim_Se

Fim_Método

Fim_Classe

McCabe pode ser aplicado para planejar os casos de testes, passando apenas uma vez em cada trecho da lógica da aplicação, minimizando os custos de testes!

SOLUÇÃO

Classe de Objetos MinMax1

inteiro n1, n2;

1 Método Comp_Numeros()

2 Início_Método

3 Exibir_mensagem "Entre com o primeiro inteiro positivo: "

4 n1 = Digitar_Valor

5 Exibir_mensagem "Entre com o segundo inteiro positivo: "

6 n2 = Digitar_Valor

7 Se n1 > 0 ou n2 > 0

8 Se (n1 > n2)

9 Mens = Concatenar("O menor numero é: ", n2)

10 Exibir_mensagem Mens;

11 Senão

12 Mens = Concatenar("O menor numero é: ", n1)

13 Exibir_mensagem Mens

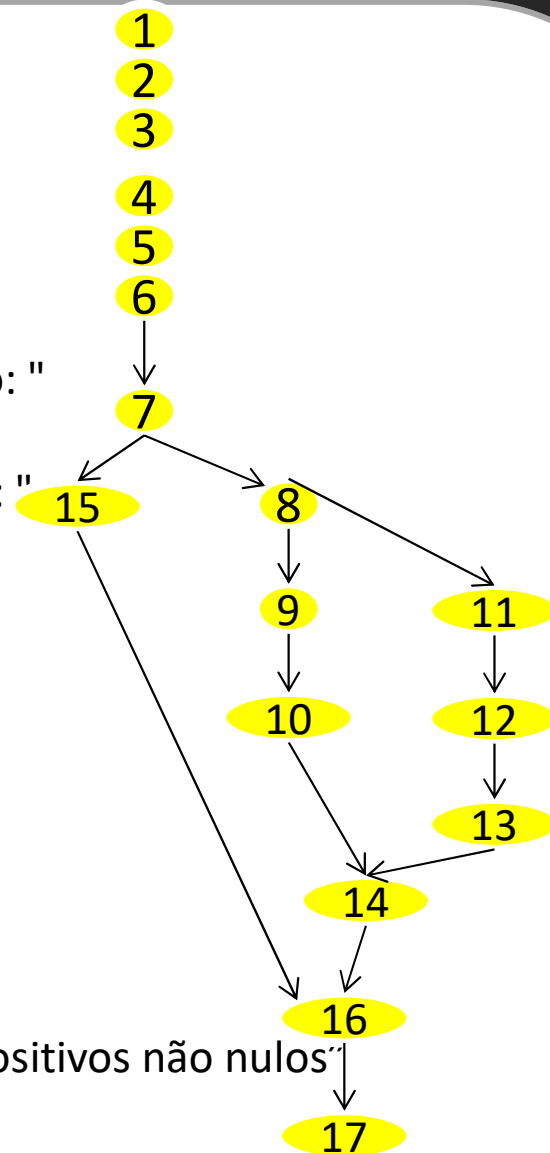
14 Fim_Se

15 Senão Exibir_mensagem "Informe somente números positivos não nulos"

16 Fim_Se

17 Fim_Método

Fim_Classe



Identificação: Teste-Compara1o NumeroMenorQue2oNumero

Nível de Teste: **Unitário**

Tipo de Teste: **Funcional**

Técnica de Teste: **Caixa PRETA**

Descrição de Objetivo: Avaliar o primeiro parâmetro de comparação de dois números como o menor deles

Programa avaliado: Método XPTO01 da Classe MinMax1

Programas associados: -

Estruturas de dados associadas: -

Entradas a fornecer:

Saídas esperadas:

n1 = 1
n2 = 3

Mensagem “O menor número é: 1”

Procedimento de teste:

Resultado esperado:

1º) O usuário executa a aplicação

1º) Mensagem “Entre com o primeiro inteiro positivo: “
exibida

2º) O usuário informa 1 em n1

2º) Mensagem “Entre com o segundo inteiro positivo: “
exibida

3º) O usuário informa 3 em n2

3º) Mensagem “O menor número é: 1” exibida