



EDUCAÇÃO E TREINAMENTO
 PARA ALAVANCAR A SUA CARREIRA

Curso e-learning
TESTER FOUNDATION

Atualizado para o
 Syllabus 2018



**Preparatório para o exame CTFL
 (Certified Tester Foundation Level)
 do ISTQB/BSTQB**

**Formação essencial de
 analistas em teste de software**




Todos os direitos de cópia reservados. Não é permitida a distribuição física ou eletrônica deste material sem a permissão expressa do autor.

Versão: 1.0 Liberação: 26/11/18

Módulo 4: Técnicas de teste					
Módulo 4 – Técnicas de teste					
CONTEÚDO PROGRAMÁTICO					
Fundamentos de teste	O teste durante todo o ciclo de vida do software	Teste estático	Técnicas de teste	Gerenciamento do teste	Ferramentas de suporte ao teste
O que é teste?	Modelos de ciclo de vida	Noções básicas	Identificando condições e casos de teste	Organização de teste	Considerações sobre ferramentas
Por que o teste é necessário?	Níveis de teste	Processo de revisão	Categorias de técnicas	Planejamento e estimativa de teste	Uso eficaz de ferramentas
Os 7 princípios do teste	Tipos de teste		Técnicas caixa-preta	Monitoramento e controle dos testes	
Processos de teste	Teste de manutenção		Técnicas caixa-branca	Gerenciamento configurações	
A psicologia do teste			Técnicas baseadas na experiência	Riscos e testes	
				Gerenciamento de defeitos	

Módulo 4: Técnicas de teste

Objetivos de aprendizagem do módulo



4.3 Técnicas de teste caixa-branca

- FL-4.3.1 (K2) Explicar a cobertura de instruções.
- FL-4.3.2 (K2) Explicar a cobertura de decisão.
- FL-4.3.3 (K2) Explicar o valor da instrução e a cobertura da decisão.

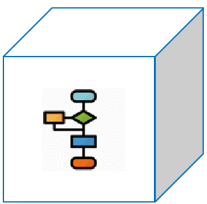
TI.exames

© Todos os direitos reservados. Proibida a redistribuição deste material.


Slide 3

Módulo 4: Técnicas de teste

Técnicas de teste caixa-branca



- São baseados na **estrutura interna** do objeto de teste.
- Geralmente usadas durante o teste de componente e de integração.
- Servem para dois propósitos:
 - Medir a cobertura do teste;
 - Desenhar casos de testes estruturais.
- Requerem que o testador entenda a **estrutura do código** e frequentemente que conheça o básico da linguagem de programação.



O testador não necessariamente precisa estar apto a escrever o código, mas precisa conhecer o básico de lógica de programação.

TI.exames

© Todos os direitos reservados. Proibida a redistribuição deste material.

Slide 4

Módulo 4: Técnicas de teste

Exemplo básico de pseudocódigo

Informe A

Informe B

Enviar

➔

```

Read A
Read B
IF A > 0 THEN
    IF B = 0 THEN
        Print "0"
    ELSE
        Print B
        IF A > 21 THEN
            Print A
        END IF
    END IF
END IF
        
```

Tl.exames © Todos os direitos reservados. Proibida a redistribuição deste material. > Slide 5

Módulo 4: Técnicas de teste

Exemplo básico de pseudocódigo

```

Read A
Read B
IF A > 0 THEN
    IF B = 0 THEN
        Print "0"
    ELSE
        Print B
        IF A > 21 THEN
            Print A
        END IF
    END IF
END IF
    
```

Executado quando A for maior que zero.

Executado quando B for igual a zero.

Executado quando B não for igual a zero.

Tl.exames © Todos os direitos reservados. Proibida a redistribuição deste material. > Slide 6

Módulo 4: Técnicas de teste

Exemplo básico de pseudocódigo

Informe A

Informe B

Enviar

➔

```

01. Read A
02. Read B
03. IF A > 0 THEN
04.   IF B = 0 THEN
05.     Print "0"
06.   ELSE
07.     Print B
08.     IF A > 21 THEN
09.       Print A
10.     END IF
11.   END IF
12. END IF

```

Caso de teste 1: A = 5; B = 0;
 Linhas executadas:
 01, 02, 03, 04, 05, 11 e 12

Caso de teste 2: A = 30; B = 5;
 Linhas executadas:
 01, 02, 03, 04, 06, 07, 08, 09, 10, 11 e 12

TI.exames © Todos os direitos reservados. Proibida a redistribuição deste material. > Slide 7

Módulo 4: Técnicas de teste

Instrução (statement) e decisão (decision)

```

Read A
Read B
IF A > 0 THEN
  IF B = 0 THEN
    Print "0"
  ELSE
    Print B
    IF A > 21 THEN
      Print A
    END IF
  END IF
END IF

```

**Instrução/Comando
(statement)**

Linhas de código que dá um comando ao programa.

**Decisão
(decision)**

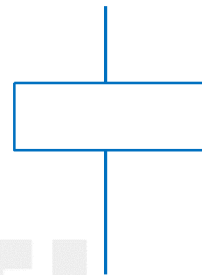
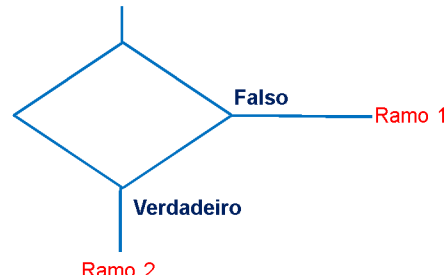
Linhas de código que avalia uma decisão a ser tomada.

TI.exames © Todos os direitos reservados. Proibida a redistribuição deste material. > Slide 8

Módulo 4: Técnicas de teste

Fluxograma é uma outra forma de representar códigos

- É muito útil desenhar um fluxograma para representar o pseudocódigo.
- Cada instrução é representada dentro de um **retângulo**.
- Cada condicional é representada dentro de um **losango**, indicando que há mais de um caminho que pode ser percorrido.

Instrução
(statement)Decisão
(decision)

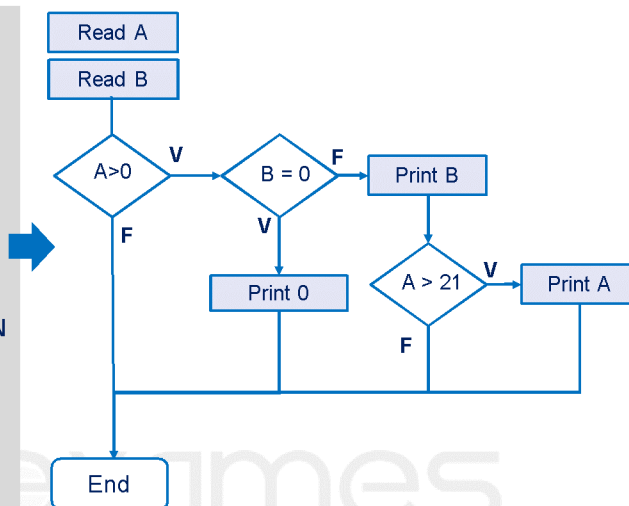
Módulo 4: Técnicas de teste

Exemplo de pseudocódigo em forma de fluxograma

```

Read A
Read B
IF A > 0 THEN
  IF B = 0 THEN
    Print "0"
  ELSE
    Print B
    IF A > 21 THEN
      Print A
    END IF
  END IF
END IF

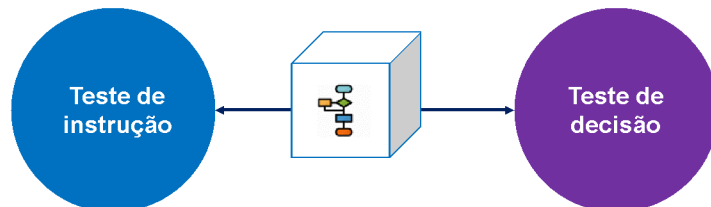
```



Módulo 4: Técnicas de teste

Técnicas caixa-branca comumente usadas

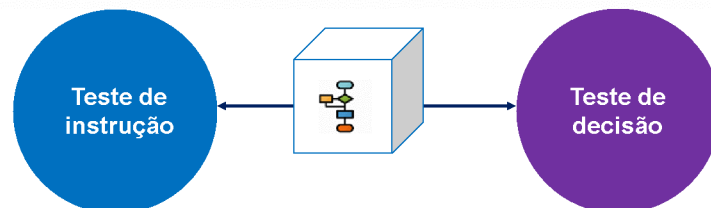
Técnicas caixa-branca são usadas em todos os níveis de teste, mas as duas técnicas relacionadas a códigos discutidas nesta seção são as mais comumente utilizadas no nível de **teste de componente**.



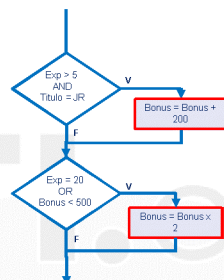
- Os casos de teste são projetados para executar instruções (comandos).
- É a técnica de caixa-branca mais fraca.
- Casos de teste são projetados para executar os resultados da decisão.
- Esta técnica de caixa-branca é mais forte.

Módulo 4: Técnicas de teste

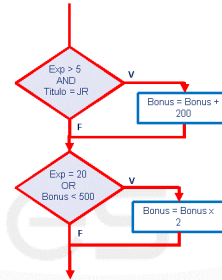
Técnicas caixa-branca comumente usadas



Cada instrução (comando) é executada pelo menos uma vez.



Cada ramo/decisão é executado pelo menos uma vez.



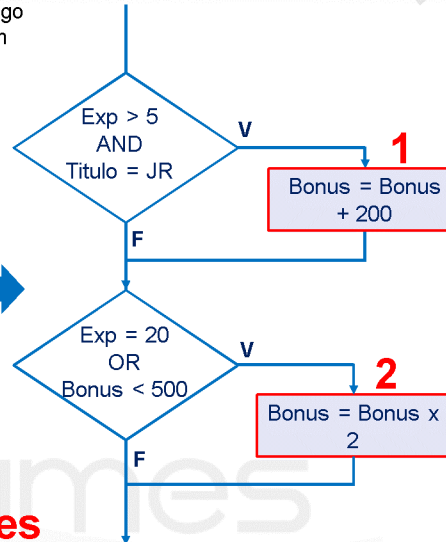
Módulo 4: Técnicas de teste

Reconhecendo as instruções no código

Abaixo temos um exemplo de fragmento de código para considerar se o funcionário tem direito a um bônus de acordo com a sua experiência.

```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF
```

```
IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```



Temos aqui 2 instruções

TI.exames © Todos os direitos reservados. Proibida a redistribuição deste material.

Slide 13

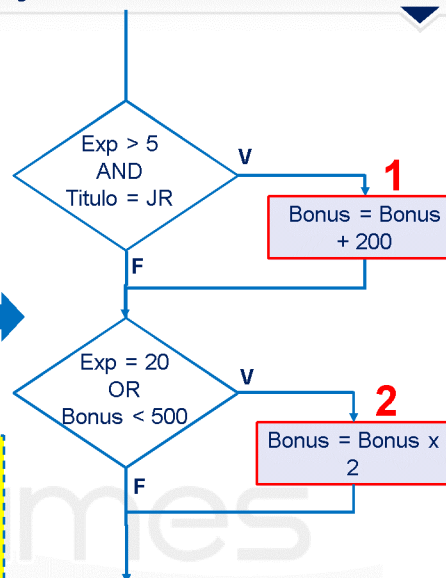
Módulo 4: Técnicas de teste

Como exercitar todas as instruções?

Exemplo de fragmento de código:

```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF
```

```
IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```



Para exercitar todas as 2 instruções de uma vez. Quais valores usar para variáveis Exp, Titulo e Bonus?

Proibida a redistribuição deste material.

Slide 14

Módulo 4: Técnicas de teste

Como exercitar todas as instruções?

Exemplo de fragmento de código:

```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF

IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```

Para exercitar todas as instruções de uma vez, os valores podem ser:
 Exp = 20
 Titulo = JR
 Bonus = 100

Slide 15

Módulo 4: Técnicas de teste

Teste de instruções e cobertura

Exemplo de fragmento de código:

```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF

IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```

2 instruções executadas
 2 instruções no total = 100%

Slide 16

Módulo 4: Técnicas de teste

Teste de instruções e cobertura

Cobertura de instruções

É medida como o número de instruções executadas pelos testes dividido pelo número total de instruções executáveis existentes, normalmente expresso como uma percentagem.



$$\frac{\text{Número de instruções/comandos executados}}{\text{Total de instruções/comandos}} \times 100\%$$

Se testarmos cada instrução “executável”, então podemos dizer que conseguimos 100% de cobertura de instrução.

Módulo 4: Técnicas de teste

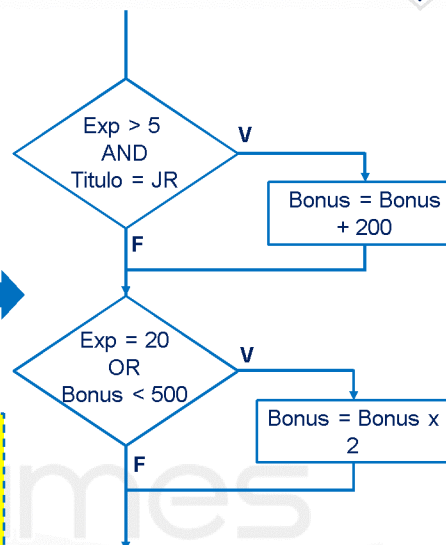
Outro exemplo para cobertura de instruções

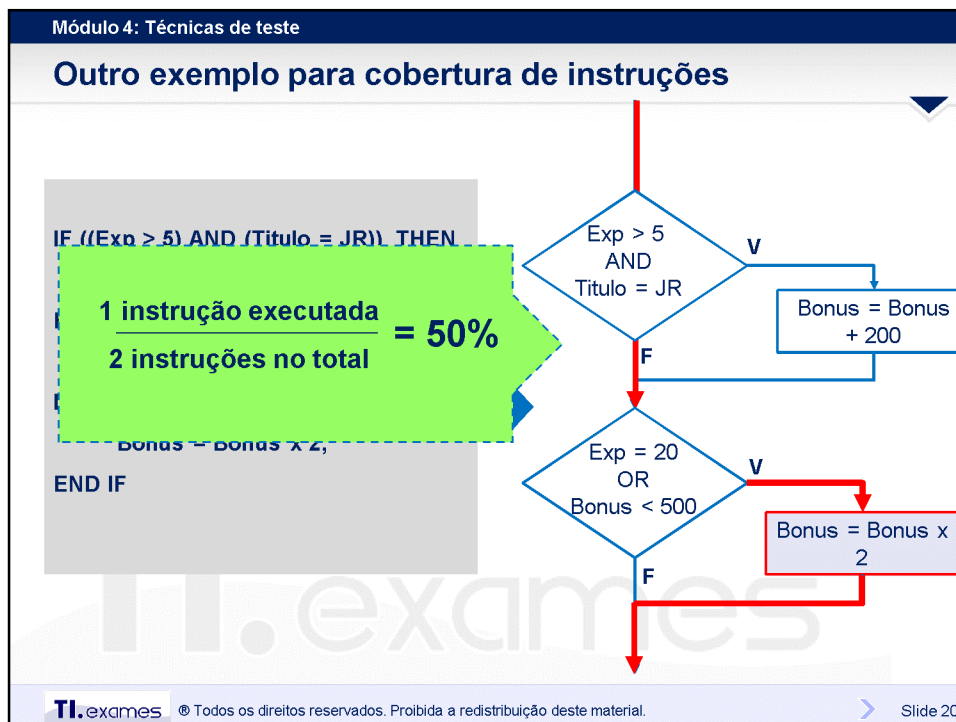
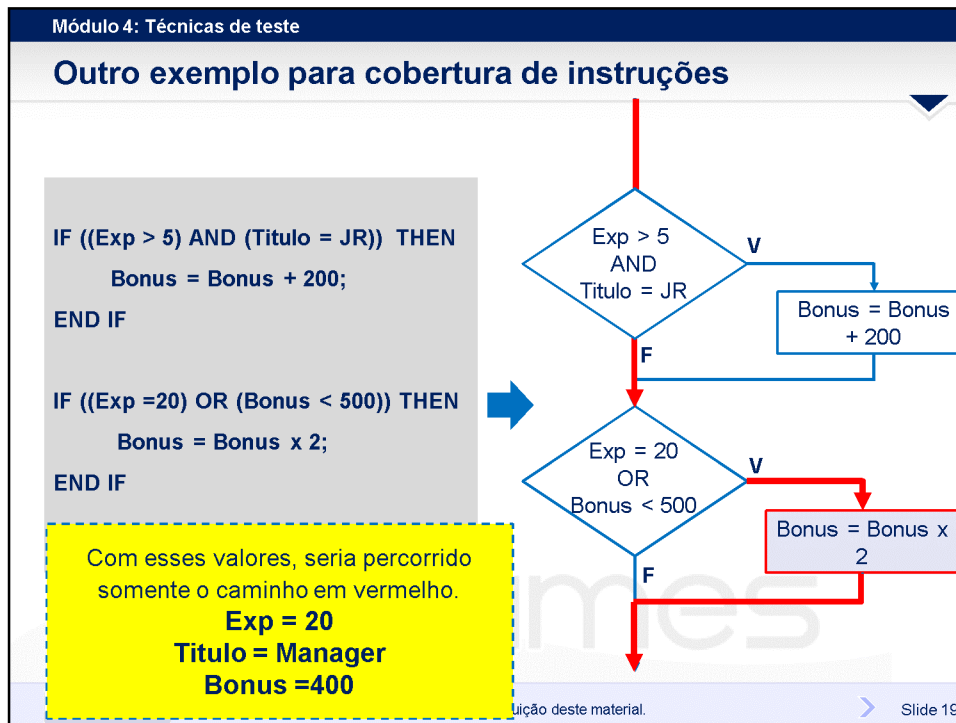
```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF
```

```
IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```

E qual seria a cobertura se os valores fossem estes abaixo?

Exp = 20
Titulo = Manager
Bonus = 400





Módulo 4: Técnicas de teste

Teste de decisão e cobertura

- Esta técnica testa as decisões existentes no código e o código executado com base nos resultados da decisão.
- Para fazer isso, os casos de teste seguem os fluxos de controle que ocorrem de um ponto de decisão (p.e., para uma instrução IF, um para o resultado verdadeiro e outro para o resultado falso; para uma instrução CASE, os casos de teste seriam necessários para todos os possíveis resultados, incluindo o resultado padrão).
- A cobertura é medida como o número de resultados de decisão executados pelos testes dividido pelo número total de resultados de decisão no objeto de teste, normalmente expresso como uma porcentagem.

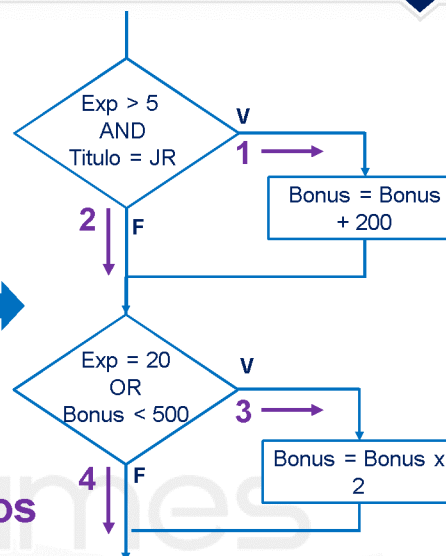
Módulo 4: Técnicas de teste

Reconhecendo os ramos/decisões na estrutura do código

```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF
```

```
IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```

Temos aqui 4 resultados de decisão (ramos).



Módulo 4: Técnicas de teste

Teste de decisão e cobertura

Cobertura de decisão

É medida como o número de resultados de decisão executados pelos testes dividido pelo número total de resultados de decisão no objeto de teste, normalmente expresso como uma porcentagem.



$$\frac{\text{Número de resultados de decisão executados}}{\text{Total de resultados de decisão}} \times 100\%$$

Se testarmos cada resultado de decisão (ramo), então podemos dizer que conseguimos 100% de cobertura de decisão.

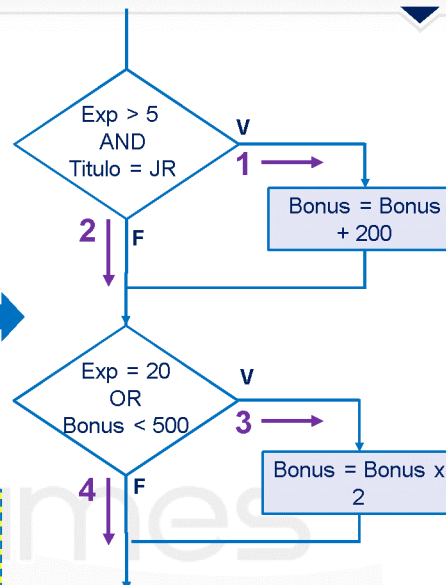
Módulo 4: Técnicas de teste

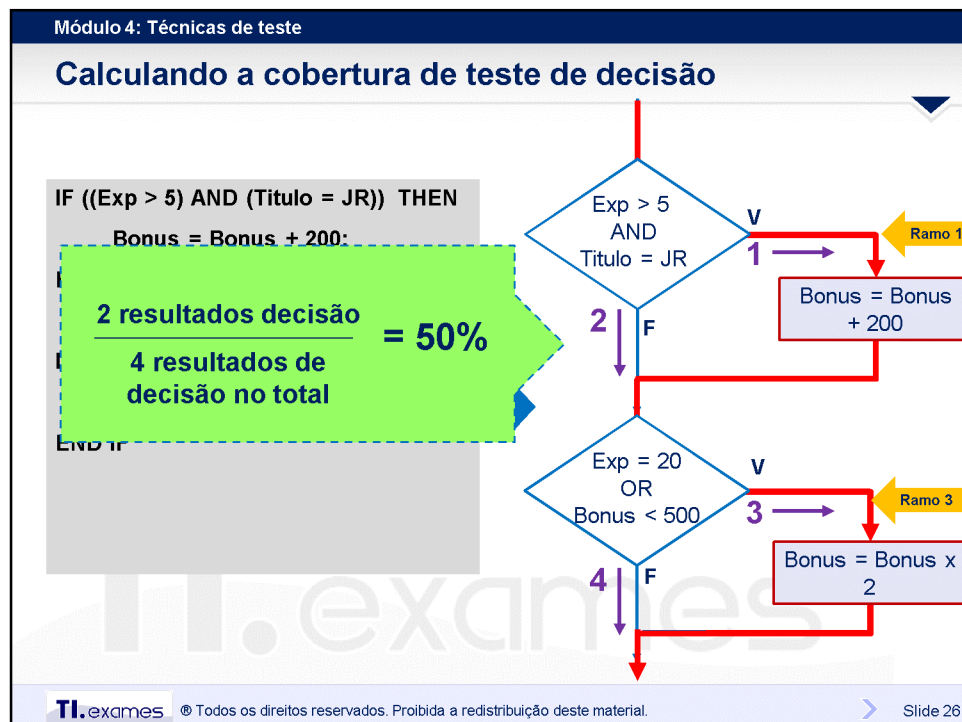
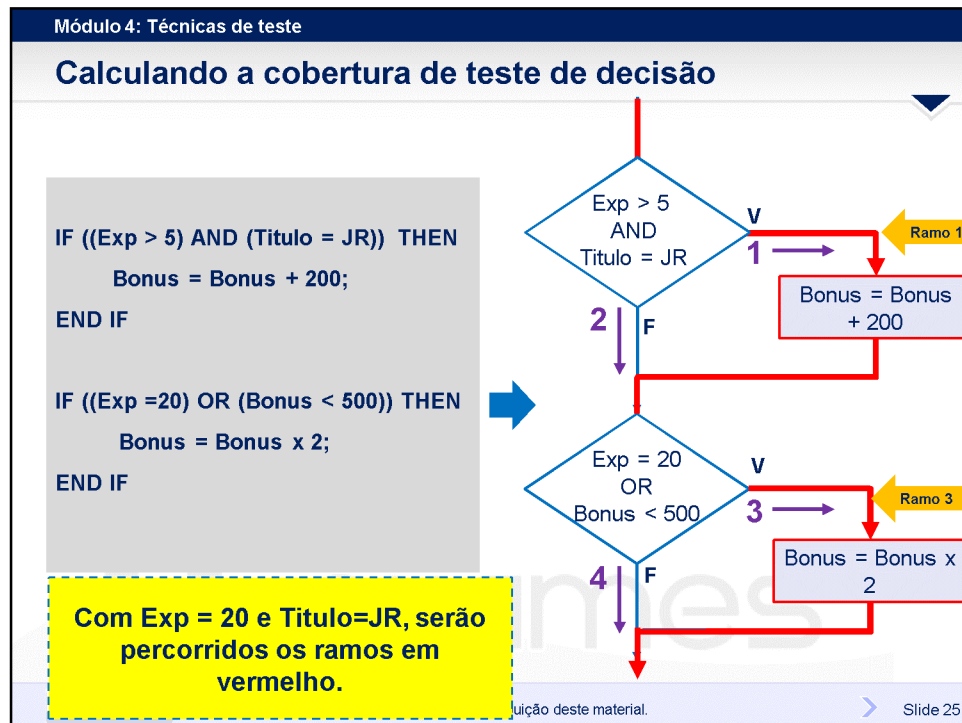
Calculando a cobertura de teste de decisão

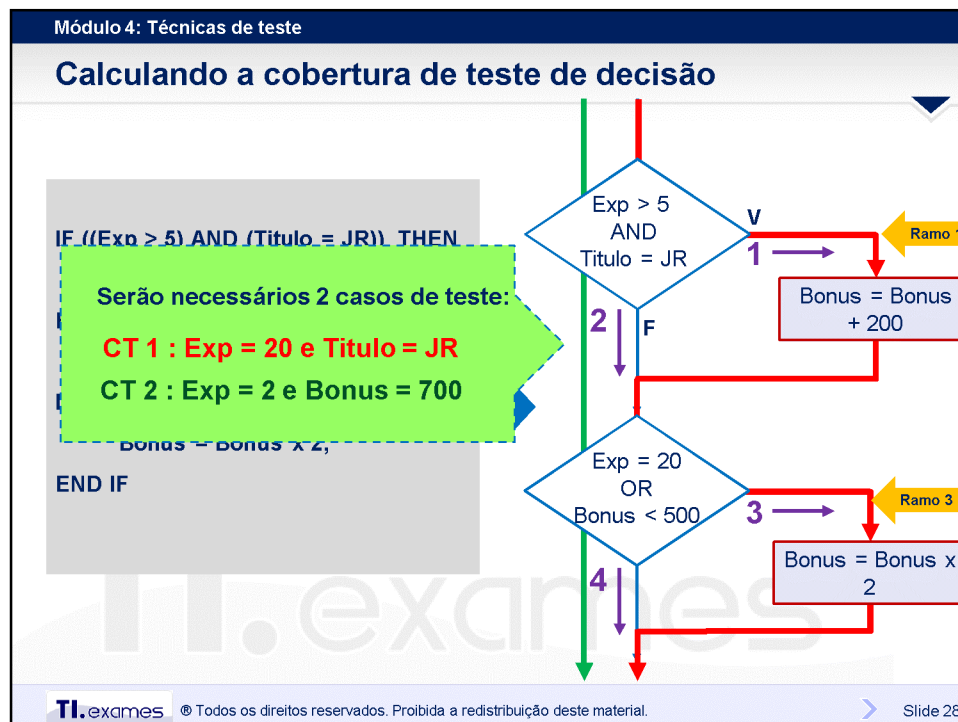
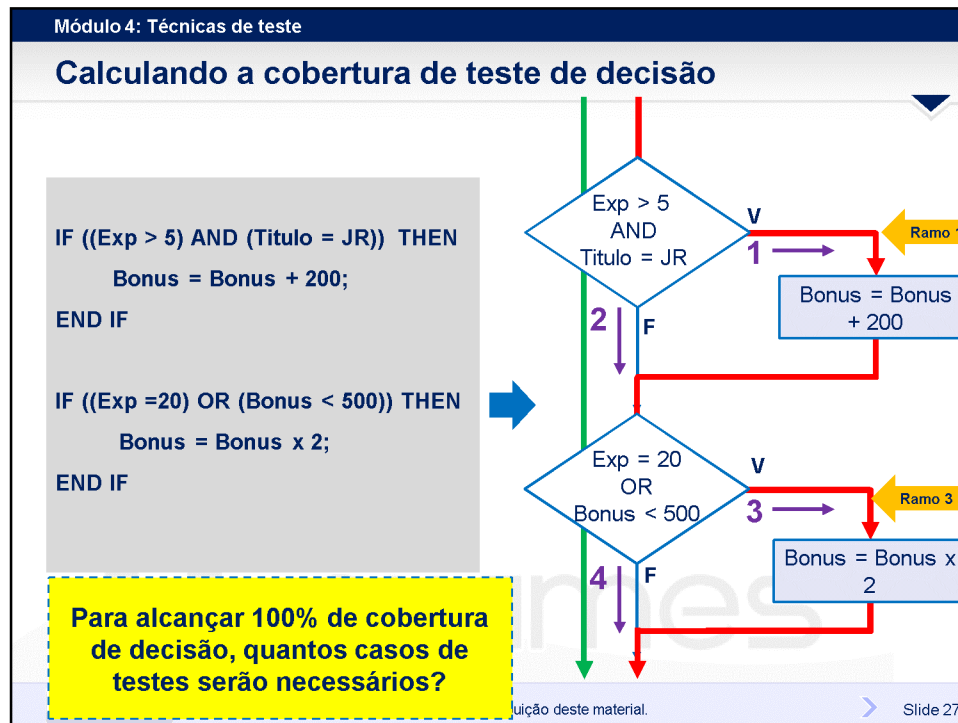
```
IF ((Exp > 5) AND (Titulo = JR)) THEN
    Bonus = Bonus + 200;
END IF
```

```
IF ((Exp = 20) OR (Bonus < 500)) THEN
    Bonus = Bonus x 2;
END IF
```

Qual será a cobertura de decisão com
Exp = 20 e Titulo=JR?







Módulo 4: Técnicas de teste

Calculando a cobertura de teste de decisão

- Quando a cobertura de 100% das instruções é alcançada, garante-se que todas as instruções executáveis no código tenham sido testadas pelo menos uma vez, mas não se garante que toda a lógica de decisão tenha sido testada.
- Das duas técnicas caixa-branca discutidas até aqui, o teste de instrução pode fornecer menos cobertura do que o teste de decisão.
- Quando uma cobertura de decisão de 100% é alcançada, representa-se que todos os resultados de decisão foram testados, o que inclui testar os resultados verdadeiro e falso, mesmo quando não há uma declaração falsa explícita (p.e., no caso de uma instrução IF sem outra no código).

Atingir 100% de cobertura de decisão garante 100% de cobertura de instrução (mas não vice-versa).

Módulo 4: Técnicas de teste

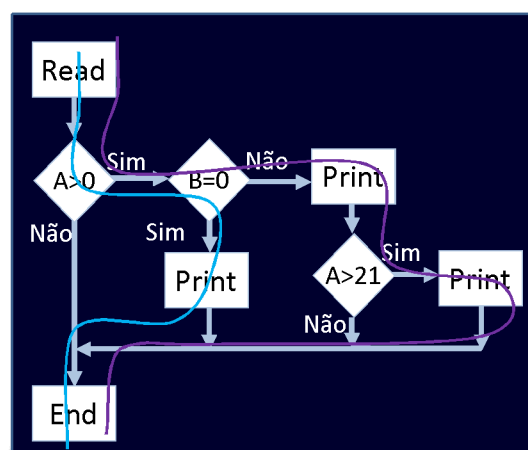
Teste de Instrução - Exemplo

Gráfico do fluxo de controle

```

1.  Read A
2.  Read B
3.  IF A > 0 THEN
4.      IF B = 0 THEN
5.          Print "Sem valor"
6.      ELSE
7.          Print B
8.          IF A > 21 THEN
9.              Print A
10.         END IF
11.     END IF
12. END IF

```

Casos de teste:

CT 1: A = 10, B = 0 | CT 2: A = 25, B = 10

Módulo 4: Técnicas de teste

Teste de Decisão - Exemplo

Gráfico do fluxo de controle

```

1. Read A
2. Read B
3. IF A > 0 THEN
4.     IF B = 0 THEN
5.         Print "Sem valor"
6.     ELSE
7.         Print B
8.         IF A > 21 THEN
9.             Print A
10.        END IF
11.    END IF
12. END IF
  
```

Casos de teste:
 CT 1: A = 0, B = 0 | CT 2: A = 10, B = 0 | CT 3: A = 10, B = 10 | CT 4: A = 25, B = 10

TI.exames © Todos os direitos reservados. Proibida a redistribuição deste material. Slide 31

Módulo 4: Técnicas de teste

Fim

Pronto, você finalizou esta seção do treinamento. Leia as instruções abaixo:

- Recomendamos neste momento fazer uma revisão dos slides para confirmar o entendimento de tudo o que foi apresentado nesta seção do treinamento.
- Após assistir à aula, recomendamos que você responda às questões do quiz clicando no botão "RESPONDER QUESTÕES DO QUIZ", disponível abaixo do vídeo da aula ou na opção "Realizar quiz" ao lado do link de cada vídeo aula.
- A sua nota obtida no quiz será exibida na lista dos módulos.

TI.exames © Todos os direitos reservados. Proibida a redistribuição deste material. Slide 32