

TESTE DE SOFTWARE

Professor Diego Oliveira
Técnica Baseada em Defeito
IFRN

CONCEITOS

- Projeto de Casos de Teste
 - Atividade on são definidos quais testes serão executados
 - Necessita de uso de técnicas para que o teste seja sistemático
 - Atividade de inteligência dos testes
- Casos de teste de defeito
 - Objetivo: encontrar os defeitos
 - O software é forçado em situações convencionais e também nas não-convencionais

CLASSIFICAÇÃO DOS TESTES (CÓDIGO FONTE)

- Caixa Branca (estrutural)
 - Com conhecimento do código-fonte
 - Geralmente utilizado em testes unitários
 - Geralmente gera uma cobertura superior de código em relação ao caixa preta
- Caixa Preta (funcional)
 - Sem conhecimento do código-fonte
 - Casos de testes projetados a partir da especificação
 - Encontra erros de omissão, mas pode não ter uma cobertura razoável de código em especificações incompletas ou muito superficiais
- O ideal é utilizar os dois tipos de teste

TÉCNICAS BÁSICAS DE PROJETO DE CASOS DE TESTE

- Caixa Preta
 - Particionamento de Equivalência
 - Análise de Valor Limite
 - Baseado em Grafos (estados da aplicação)
- Caixa Branca
 - Condições Lógicas (Branch)
 - Ciclos (Loops)
 - Caminhos (Gráfico de Fluxo de Controle)

PARTICIONAMENTO DE EQUIVALÊNCIA

- Entrada de dados
 - não é possível cobrirmos o domínio completo de cada campo
 - Ex: idade de 0 a 130 = 131 casos de testes
- Divide-se o domínio em classes de equivalência
 - Faixa de valores onde o software tem o mesmo comportamento
- Ex: idade de 0 a 130, obrigatório
 - Menor que zero = inválido
 - 0 a 130 = válido
 - Maior que 130 = inválido
 - Não preenchido = inválido
- Cada classe de equivalência precisa de um caso de teste

PARTICIONAMENTO DE EQUIVALÊNCIA

- Outro exemplo: campos de texto de 1 a 255 caracteres, obrigatório
 - Mais que 255 = inválido
 - Não preenchido (0) = inválido

PARTICIONAMENTO DE EQUIVALÊNCIA

- Campo de data de nascimento, obrigatório
 - Data válida no passado = válida
 - Data válida no futuro = inválida
 - Data inexistente (31/02/1990) = inválida
 - Não preenchido = inválido

ANÁLISE DE VALOR LIMITE

- Geralmente os programadores erram expressões lógicas de validação
- Para revelar estes defeitos, precisa-se exercitar os limites
- Para cada limite, testamos o valor imediatamente anterior e o valor imediatamente posterior
- Ex: idade de 0 a 130:
 - -1, 0, 1 = três casos de teste
 - 129, 130, 131 = mais três casos de teste

ANÁLISE DE VALOR LIMITE

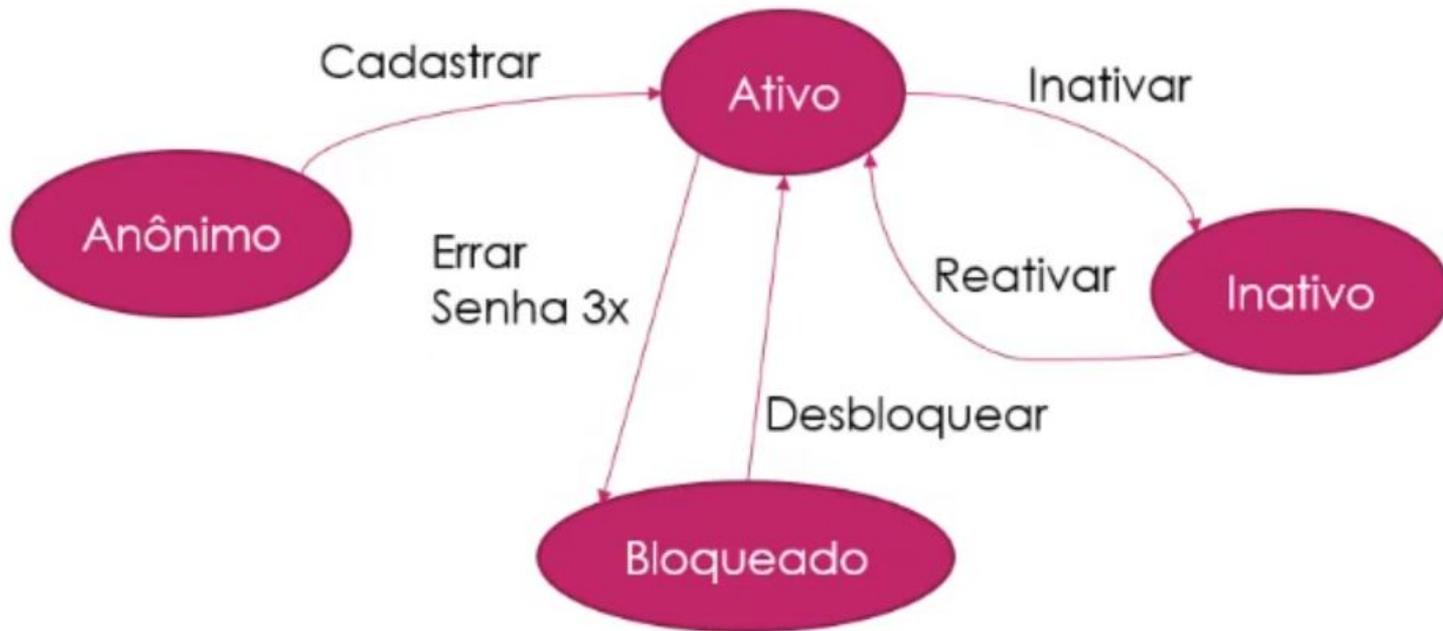
- Caracteres
 - Limites são quantidades de caracteres
- Datas
 - Depende se a especificação dá limites para as datas
 - Pode não ter limite superior ou inferior
- Float
 - Qual a precisão que se deseja?
 - Moeda = 2 dígitos

ESTADOS

- Aplicações podem ter estados, assim como componentes individuais
- Casos de Aceitação
 - Testar transições de estados válidas
- Casos de Defeitos
 - Transações inválidas
 - Loops de estados
 - Caminhos limitados
- Melhor de visualizar se tivermos um diagrama de estados

ESTADOS

- Usuário



CONDIÇÕES LÓGICAS (BRANCH) - CAIXA BRANCA

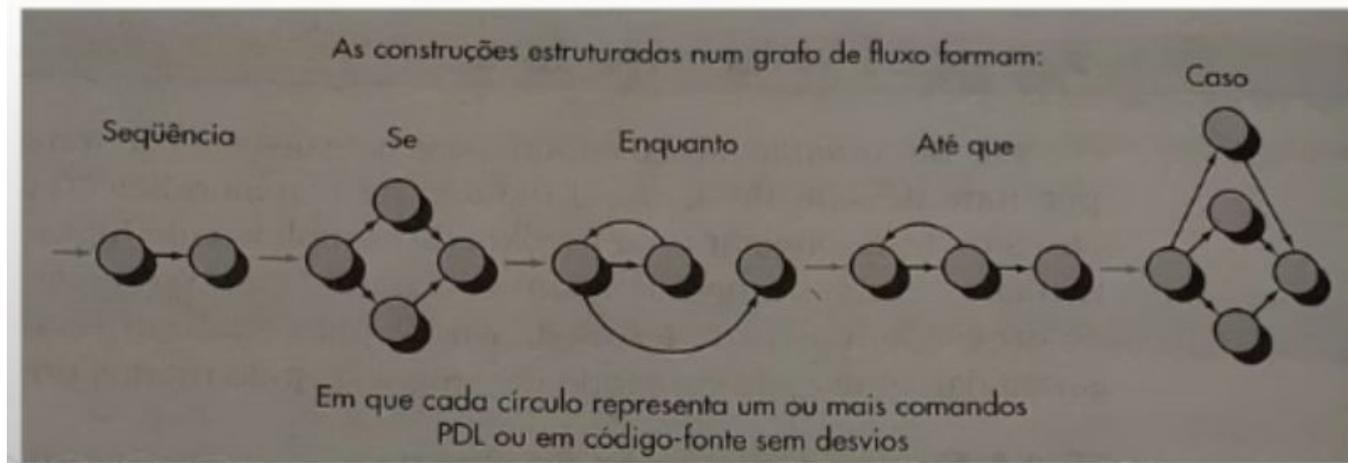
- Cada expressão lógica pode ser exercitada de diversas maneiras
- A && B (operador de curto circuito, se A é falso, então B não é avaliado)
 - Precisamos de casos de teste onde B seja avaliado, pois algum erro pode ser gerado ao avaliar B
- Coberturas
 - Garantir que todas as partes da expressão sejam avaliadas
 - Garantir que a expressão avalie para true e para false
 - Em último nível, tabela verdade, todas as combinações
(somente para softwares críticos)

CICLOS (LOOPS) - CAIXA BRANCA

- Loops formam caminhos no código e também possuem expressões lógicas
- Podem ser dependentes de estados
- Casos de teste típicos para testar loops são:
 - Não passar pelo loop
 - Passar uma vez
 - Passar mais de uma vez
 - Passar X vezes (onde X é menor que N = limite de iterações)
 - Passar N-1, N e N+1 vezes

CAMINHOS (LIMITADOS) - CAIXA BRANCA

- Mesmo testando as expressões lógicas, loops e tendo cobertura de 100% das linhas e branches, podemos não estar exercitando todos os caminhos do código
- Grafo de Fluxo de Controle ajuda a visualizar caminhos



CAMINHOS (LIMITADOS) - CAIXA BRANCA

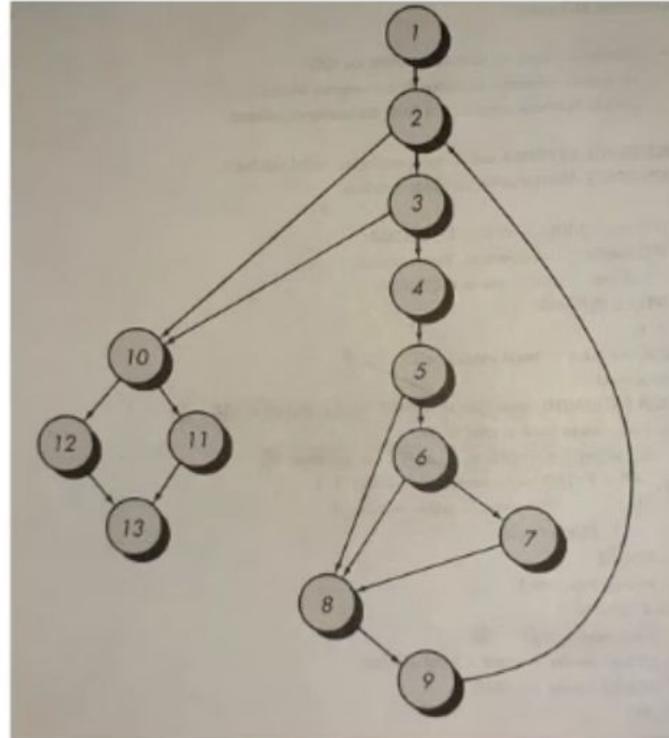
PROCEDIMENTO média:

- Este procedimento calcula a média de 100 ou menos números situados entre valores limites; calcula também a soma e o total de números válidos.

INTERFACE RETORNA média total,entradas, total,válidas;
INTERFACE ACEITA valor, mínimo, máximo:

TIPO valor[1:100] É VETOR DE ESCALA;
TIPO média, total,entradas, total,válidas;
mínimo, máximo, soma É ESCALA;
TIPO i É INTEIRO:

```
1 i = 1;  
total.entradas = total.válidas = 0;  
soma = 0;  
FAÇA ENQUANTO valor[i] <> -999 E total.entradas < 100  
4 incremente total.entradas de 1;  
SE valor[i] >= mínimo E valor[i] <= máximo  
6 ENTÃO incremente total.válidas de 1;  
7 soma = soma + valor[i];  
SENÃO pule  
8 FIM-SE  
9 incremente i de 1;  
10 FIM-ENQUANTO  
SE total.válidas > 0  
11 ENTÃO média = soma / total.válidas;  
12 SENÃO média = -999;  
13 FIM-SE  
FIM média
```



Fonte: Livro do Pressman

PERGUNTAS?

