Testando os Programas

Engenharia de Software II

INSTITUTO FEDERAL
Mato Grosso do Sul

O que é Teste de Software?

- Teste de Software é o processo de verificar se o produto está de acordo com as especificações funcionando corretamente conforme foi projetado.

- O teste é destinado a mostrar que um programa faz o que é proposto a fazer e para descobrir os defeitos do programa antes do uso.

INSTITUTO FEDERAL
Mato Grosso do Sul

O que é Teste de Software?

- Teste é o processo de operar um sistema ou componente do sistema sob condições específicas, observando e registrando o resultado e fazendo uma avaliação de alguns aspectos do sistema ou componente - *IEEE Standard Glos*sary

- Além disso, o teste é uma parte importante no processo de integração contínua de um software, pois ajuda a evitar erros em cada atualização de um sistema

INSTITUTO FEDERAL
Mato Grosso do Sul

Por que testar?

- O teste não é a primeira etapa para encontrar defeitos. É importante fazer uma revisão dos requisitos e do projeto no começo do desenvolvimento.

- Quem dera se não cometêssemos erros e que todo programa funcionasse 100%.

- Desenvolver software é complexo, pois lida com um grande número de estados e com fórmulas, atividades e algoritmos complexos.

INSTITUTO FEDERAL
Mato Grosso do Sul

Por que testar?

- Além disso, utilizamos as ferramentas disponíveis para implementar as concepções de um sistema, quando o cliente pode estar inseguro quanto ao que realmente quer.

- quanto mais cedo achar um defeito mais barato ele vai custar

INSTITUTO FEDERAL
Mato Grosso do Sul

Objetivos do processo de teste de software

1. Demonstrar ao desenvolvedor e ao cliente que o software atende seus requisitos.
   - Para softwares customizados deve haver pelo menos um teste para cada requisito
   - Para softwares genéricos deve haver testes para todas as características do sistema

2. Descobrir situações que o software se comporta de maneira incorreta, indesejável ou de forma diferente das especificações. Ex. panes, interações indesejáveis com outros sistemas, processamento incorreto e corrupção de dados.

INSTITUTO FEDERAL
Mato Grosso do Sul

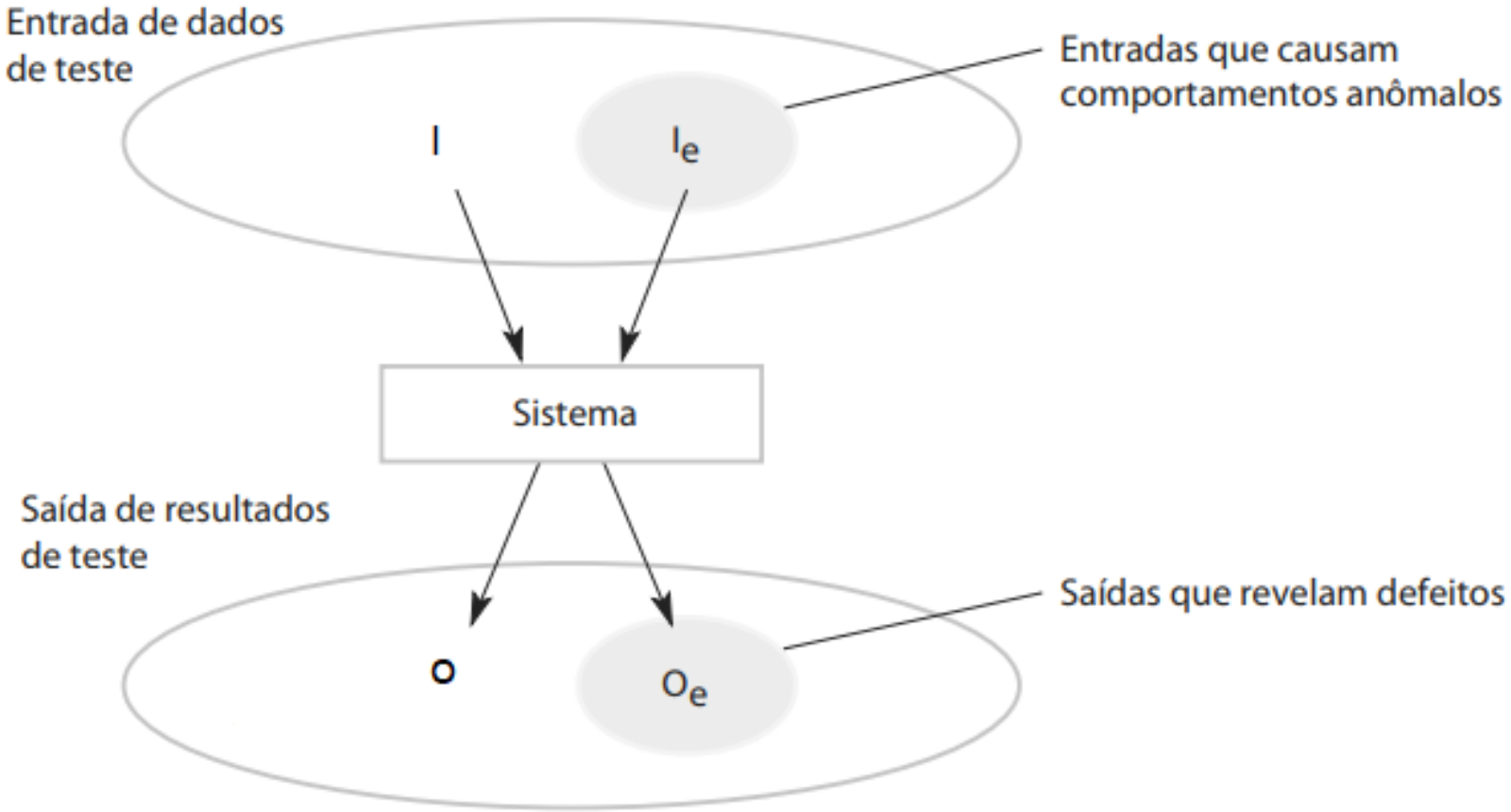
Objetivos do processo de teste de software

- O primeiro objetivo leva a **testes de validação**, onde você espera que o sistema execute corretamente usando determinado conjunto de casos de teste que reflete o uso esperado do sistema.

- O segundo objetivo leva a **testes de defeitos**, nos quais os casos de teste são projetados para expor os defeitos. Estes não precisam refletir com precisão a maneira como o sistema costuma ser usado.

INSTITUTO FEDERAL
Mato Grosso do Sul

Um modelo de entrada-saída de teste de programa



INSTITUTO FEDERAL
Mato Grosso do Sul

Verificação e Validação (V&V)

- Os testes podem mostrar apenas presenças de erros e não ausência (DIJKSTRA et al. 1972).

- O teste é parte de um amplo processo de verificação e validação:
  - Validação: estamos construindo o produto certo?
  - Verificação: estamos construindo o produto da maneira certa?

BOEHM, 1979.

INSTITUTO FEDERAL
Mato Grosso do Sul

Verificação e Validação (V&V)

- VALIDAÇÃO: o objetivo da validação é garantir que o software atenda às expectativas do cliente.
- VERIFICAÇÃO: o objetivo da verificação é checar se o software atende a seus requisitos funcionais e não funcionais.
- O objetivo final dos processos de verificação e validação é estabelecer a confiança de que o software está pronto para seu propósito. Isso significa que o sistema deve ser bom o suficiente para seu intuito.

INSTITUTO FEDERAL
Mato Grosso do Sul

Quando devo iniciar os testes?

- Em todo o ciclo de vida de desenvolvimento do software

INSTITUTO FEDERAL
Mato Grosso do Sul

Quando devo iniciar os testes?

- Em todo o ciclo de vida de desenvolvimento do software
- Veremos na próxima aula que o TDD testa primeiro e codifica depois.

INSTITUTO FEDERAL
Mato Grosso do Sul

Defeitos e Falhas de Software

- Defeito ⇨ Erro ⇨ Falha
  - Defeito: deficiência mecânica ou algorítmica que, se ativada, pode levar a uma falha
  - Erro: item de informação ou estado de execução inconsistente
  - Falha: evento notável em que o sistema viola suas especificações

INSTITUTO FEDERAL
Mato Grosso do Sul

Defeitos e Falhas de Software

- Defeito ⇒ Erro ⇒ Falha
  - Defeito: deficiência mecânica ou algorítmica que, se ativada, pode levar a uma falha
  - Erro: item de informação ou estado de execução inconsistente
  - Falha: evento notável em que o sistema viola suas especificações

bug

INSTITUTO FEDERAL
Mato Grosso do Sul

Defeitos e Falhas de Software

Segundo o padrão IEEE (1990), *defeito (fault)* é um passo, processo ou definição de dados incorreto, por exemplo, uma instrução ou comando incorreto no programa; *engano (mistake)* é uma ação humana que produz um resultado incorreto, por exemplo, uma ação incorreta feita pelo programador; *erro (error)* ocorre quando o valor esperado e o valor obtido não são os mesmos, ou seja, qualquer resultado inesperado na execução do programa constitui um erro e *falha (failure)* é a produção de uma saída incorreta em relação à especificação. Neste texto, são utilizados os termos erro (causa) e falha (conseqüência), sendo que o termo erro engloba os termos defeito, engano e erro.

INSTITUTO FEDERAL
Mato Grosso do Sul

Defeitos e Falhas de Software

- A falha é o resultado de um ou mais defeitos:
  - A especificação pode estar errada ou falta um requisito.
  - A especificação pode conter um requisito impossível de se implementar, considerando o hardware e o software estabelecidos.
  - O projeto do sistema pode conter um defeito. Talvez os projetos do banco de dados e da linguagem de consulta tornem impossível autorizar os usuários.
  - O projeto do programa pode conter um defeito.
  - O código de programa (algoritmo) possa estar errado.

INSTITUTO FEDERAL
Mato Grosso do Sul

Defeitos e Falhas de Software

- A identificação de defeito é o processo de determinar qual defeito ou defeitos causaram a falha
- E a correção ou eliminação de defeito é o processo de fazer mudanças no sistema de modo que os defeitos sejam eliminados.
- Assim, as práticas de engenharia de software servem para controlar a qualidade do código que escrevemos.

INSTITUTO FEDERAL
Mato Grosso do Sul

Tipos de defeitos

- **Os principais tipos são:**
- **D. no algoritmo:**
  - Desvios muito antecipado
  - Desvio muito tardios
  - Teste de condição errada
  - Esquecer de inicializar as variáveis ou de definir invariantes do loop
  - Esquecer de testar uma condição em particular (i.g. divisão por zero)
  - Comparar variáveis de tipos de dados inadequados

INSTITUTO FEDERAL
Mato Grosso do Sul

Tipos de defeitos

- **Os principais tipos são:**
- **D. de sintaxe:**
  - Usar corretamente a linguagem de programação
- **D. de computação e de precisão:**
  - Fórmula errada ou não calcula o resultado com o grau de precisão requerido.
- **D. na documentação:**
  - Documentação diverge do que o programa realmente faz.
- **D. por sobrecarga (Stress):**
  - Limite do comprimento das filas, buffers, tabelas, etc.

INSTITUTO FEDERAL
Mato Grosso do Sul

Tipos de defeitos

- **Os principais tipos são:**
- **D. de capacidade ou a limites:**
  - Se foi projetado para gerenciar 32 dispositivos, não pode colocar 33
  - Número de acesso a disco, número de interrupções, número de tarefas executadas simultaneamente, etc.
- **D. de sincronia (ou de coordenação):**
  - Aplica-se para sistemas de tempo-real
  - Código que gerencia os eventos é inadequado

INSTITUTO FEDERAL
Mato Grosso do Sul

Tipos de defeitos

- **Os principais tipos são:**
- **D. de desempenho ou ‘throughput’:**
  - Ocorre quando o sistema não funciona com a velocidade prescrita pelos requisitos
  - Neste defeito, as restrições de tempo são adicionadas pelos requisitos do cliente
- **D. de recuperação:**
  - Quando se tem uma falha e o sistema não se comporta normalmente

INSTITUTO FEDERAL
Mato Grosso do Sul

Tipos de defeitos

- **Os principais tipos são:**
- **D. do hardware e do sistema:**
  - Quando não funcionam de acordo com as condições e os procedimentos operacionais documentados.
- **D. de padrões e procedimentos:**
  - Pode afetar o ambiente de desenvolvimento
  - Quando um programador deixa de seguir os padrões exigidos, dificulta para outro programador solucionar um problema.

INSTITUTO FEDERAL
Mato Grosso do Sul

Referências

- Testes de Software: https://www.devmedia.com.br/testes-de-software-introducao/22281

- Engenharia de Software: Ian Sommerville, 2011.

- Teste de Software: https://www.devmedia.com.br/teste-de-software/
- https://www.devmedia.com.br/testes-de-software-introducao/22281