Tópicos Avançados A
(020290-A)

Aula 01: Fundamentos de Teste de Software

Prof. Fabiano Cutigi Ferrari

1º semestre 2015

fabiano@dc.ufscar.br

LaPES – Laboratório de Pesquisa em Engenharia de Software
Departamento de Computação – UFSCar

Sumário

- Visão geral sobre teste de software:
  - Garantia de Qualidade de Software
  - Objetivos do Teste
  - Conceitos básicos

Perguntas Iniciais

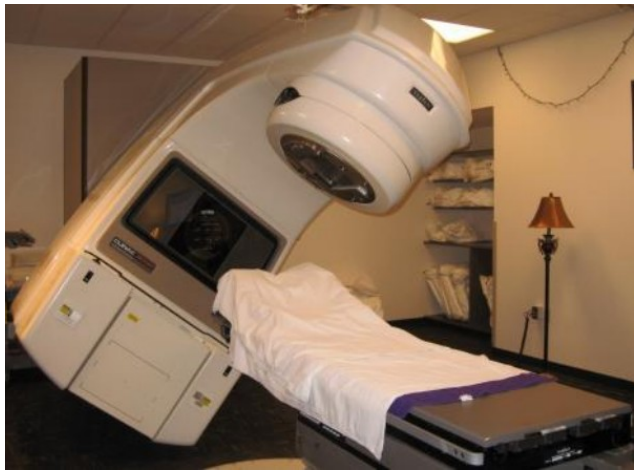
- Quem de vocês já testou um software?
- Como?
- Quem de vocês já testou **de forma sistemática** um software?

Perguntas Iniciais

- Qual é a porcentagem do tempo e do custo total de desenvolvimento de software que é, em geral, gasto com teste?
  - **R: 50% do tempo e 50% do custo**

Importância da Atividade de Teste

- 1986: THERAC-25, máquina de tratamento de radiação controlada por computador
- Dois pacientes com câncer do Centro de Câncer do Texas em Tyler receberam doses fatais de radiação
- Por que: bug – falha em condição de concorrência (tarefas concorrentes não coordenadas corretamente)



Importância da Atividade de Teste

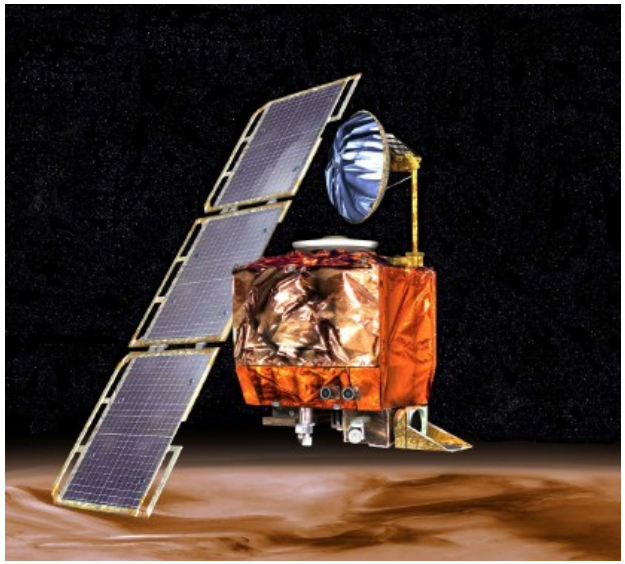
- 1992: Serviço de Despacho de Ambulâncias de Londres – propósito: automatizar processos manuais associados ao serviço de ambulâncias do Reino Unido
- Falhou em 26 e 27 de novembro de 1992
- Carga aumentou, emergências acumularam → sistema fez alocações incorretas
  - mais de uma ambulância no mesmo incidente
  - veículo mais próximo não foi alocado
- Aproximadamente 46 mortes poderiam ter sido evitadas

Importância da Atividade de Teste

- 1998: US Vincennes (míssil) derrubou um Airbus 300 da Iran Air
- Confundiu com um F-14: 290 pessoas morreram
- Por que: bug – saída “estranha” dada pelo software de rastreamento

Importância da Atividade de Teste

- 1999: Mars climate orbiter – propósito: mandar sinais da sonda Mars Polar
- Desastre: caiu no planeta em vez de atingir órbita
- Por que: bug – falha em converter unidades de medida
  - US$ 165M



Problema

- Aprende-se a programar sem aprender a testar
- MIT:
  - <https://www.koofers.com/videos/testing-and-debugging/>
- Mesmo que software não-crítico: profissionalismo... pense em uma montadora de carros

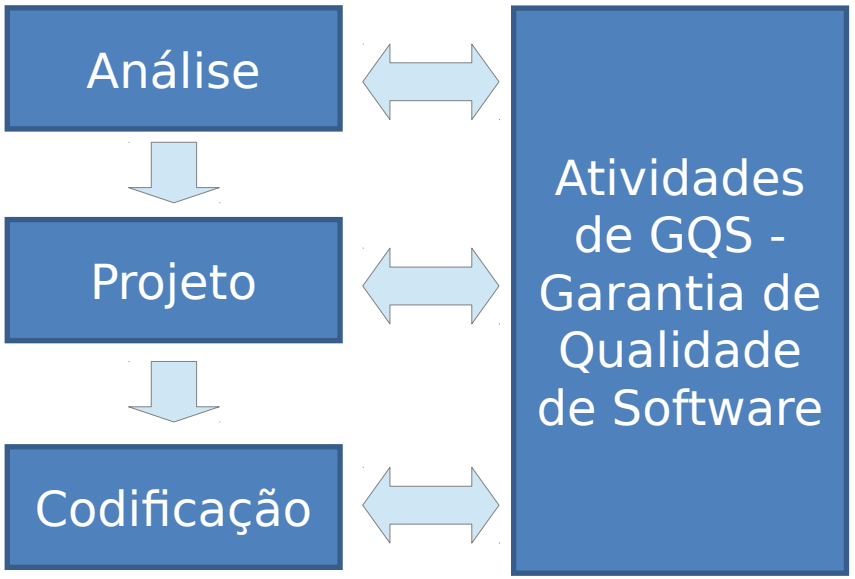
Pontos Importantes

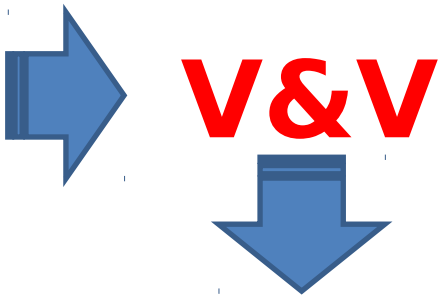
- A maior parte dos defeitos é de origem humana
- São gerados na comunicação e na transformação de informações
- Continuam presentes nos diversos produtos de software produzidos e liberados (10 defeitos a cada 1000 linhas de código)
- A maioria encontra-se em partes do código raramente executadas

Garantia da Qualidade de Software

Teste & Ciclo de Vida de Desenvolvimento

Fases Genéricas

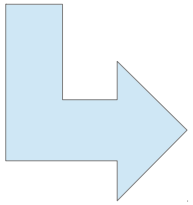




- Teste
- Inspeção
- Análise estática
- Walkthroughs
- ...

Garantia da Qualidade de Software

- É um conjunto de atividades técnicas aplicadas durante todo o processo de desenvolvimento.
- O objetivo é garantir que tanto o **processo** de desenvolvimento quanto o **produto** de software atinjam níveis de qualidade especificados.



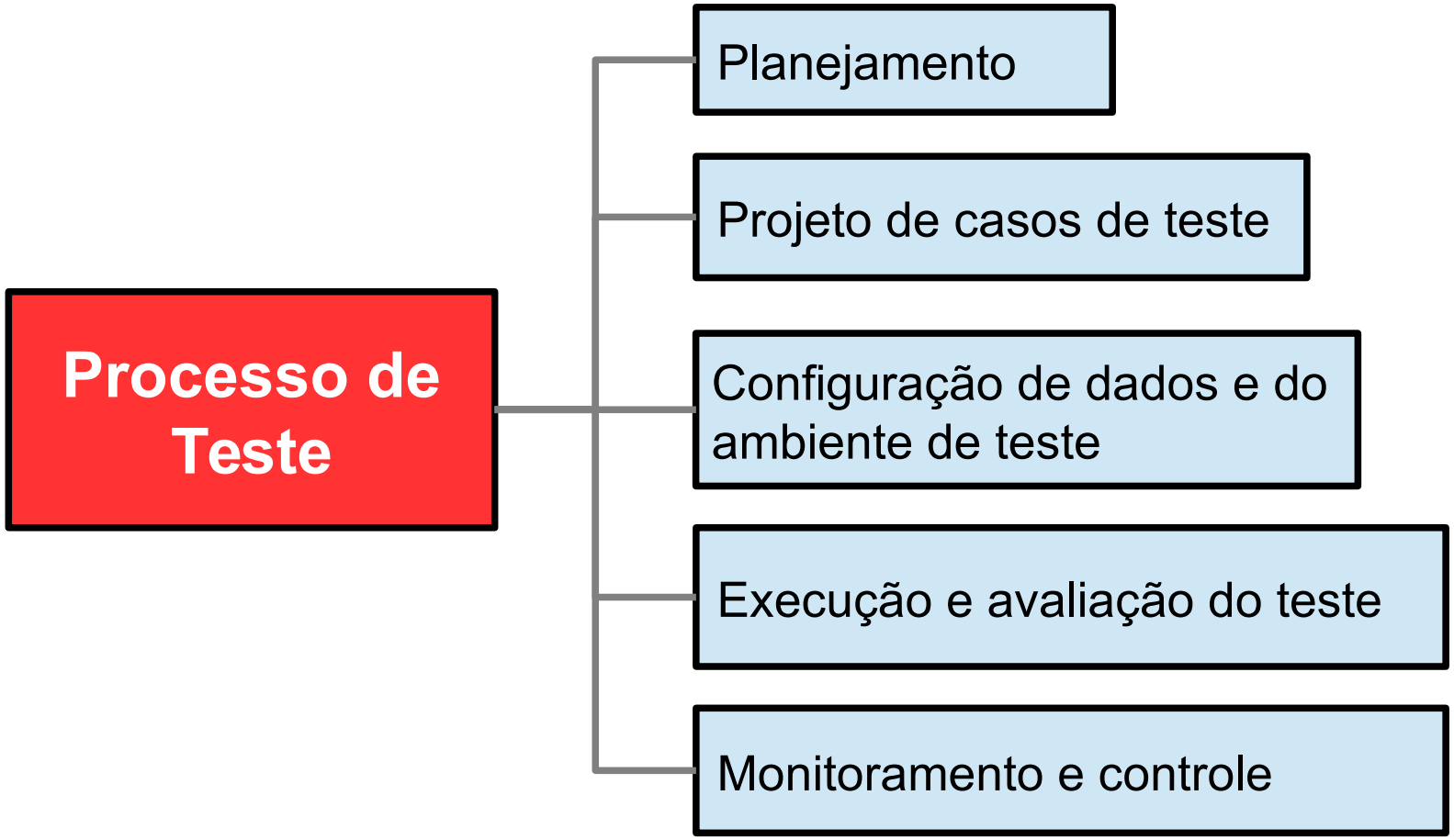
V&V: Verificação e Validação

Verificação e Validação de Software

- Verificação: Assegurar consistência, completude e corretude do produto em cada fase e entre fases consecutivas do ciclo de vida do software.
  - **"Estamos construindo corretamente o produto?"**

- Validação: Assegurar que o produto final corresponda aos requisitos do software.
  - **"Estamos construindo o produto certo?"**
  - Principal atividade: teste de software

Processo de Teste de Software



Höhn, E. N. KITest: Um arcabouço de conhecimento e melhoria de processo de teste. Tese (Doutorado) ICMC/USP, São Carlos, SP - Brasil, junho 2011.

Processo de Teste de Software

- **Planejamento**:
  - Análise de risco, definição de recursos e ambientes necessários, critérios de parada, cronograma, resultados esperados, equipe etc.

- **Projeto de casos de teste**:
  - Definição dos cenários, identificação e priorização de casos de teste, criação de casos de teste, identificar dados de teste específicos etc.

Processo de Teste de Software

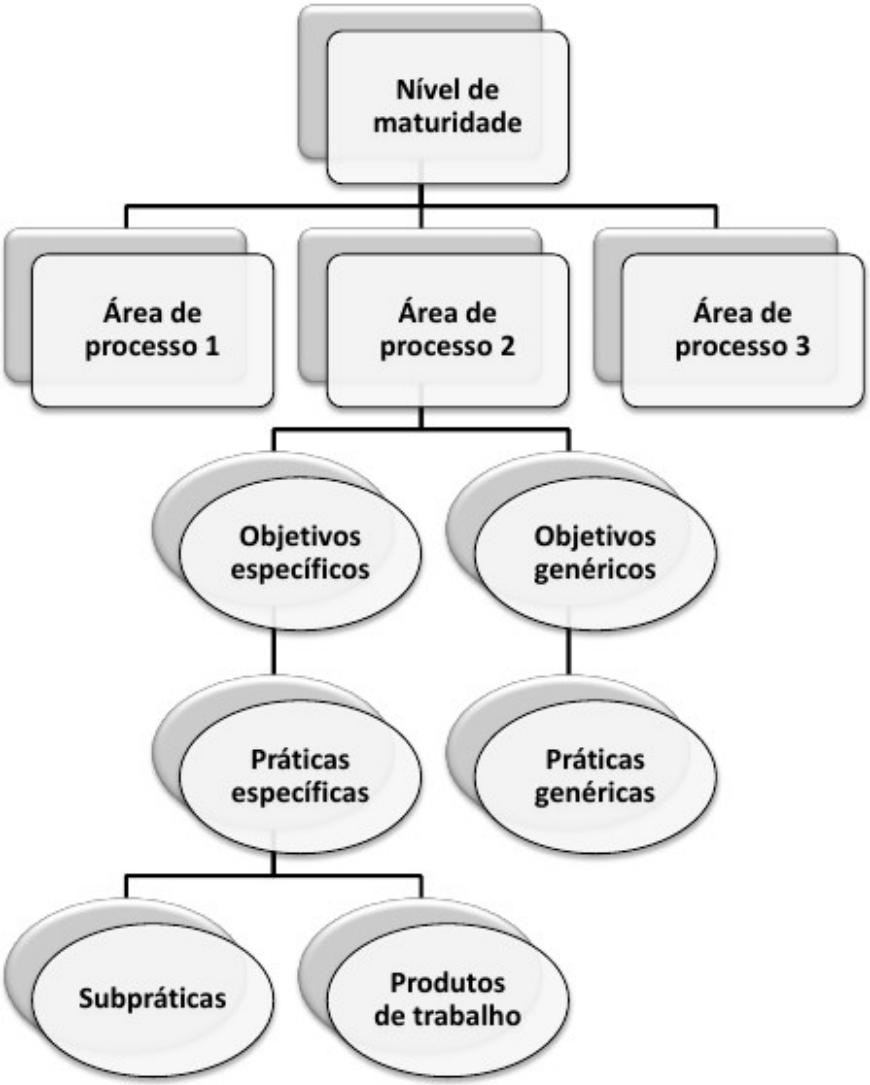
- **Configuração de dados e do ambiente de teste**:
  - Desenvolver e priorizar procedimentos de teste, implementar o ambiente de teste, realizar pré-teste etc.
- **Execução e avaliação do teste**:
  - Executar casos de teste, relatar incidentes de teste, escrever log de teste, decidir sobre incidentes de teste, acompanhar o status dos incidentes etc.
- **Monitoramento e controle**:
  - Conduzir revisões do progresso do teste, monitorar defeitos, conduzir revisões de qualidade do produto, analisar problemas, tomar ações corretivas etc.

TMMi – Test Maturity Model integration

- Inspirado no CMMI
- Modelo de maturidade especializado na melhoria do processo de teste
- 5 níveis de maturidade

TMMi – Test Maturity Model integration

- Cada nível de maturidade é composto por Áreas de Processo (*Process Area - PA*)
- Cada Área de Processo é formada por Objetivos Específicos (*Specific Goals - SG*)
- Cada Objetivo Específico é formado por Práticas Específicas (Specific Practices - SP)
- Há também Objetivos Genéricos (e respectivas Práticas Genéricas), que podem se relacionar com mais de uma PA.

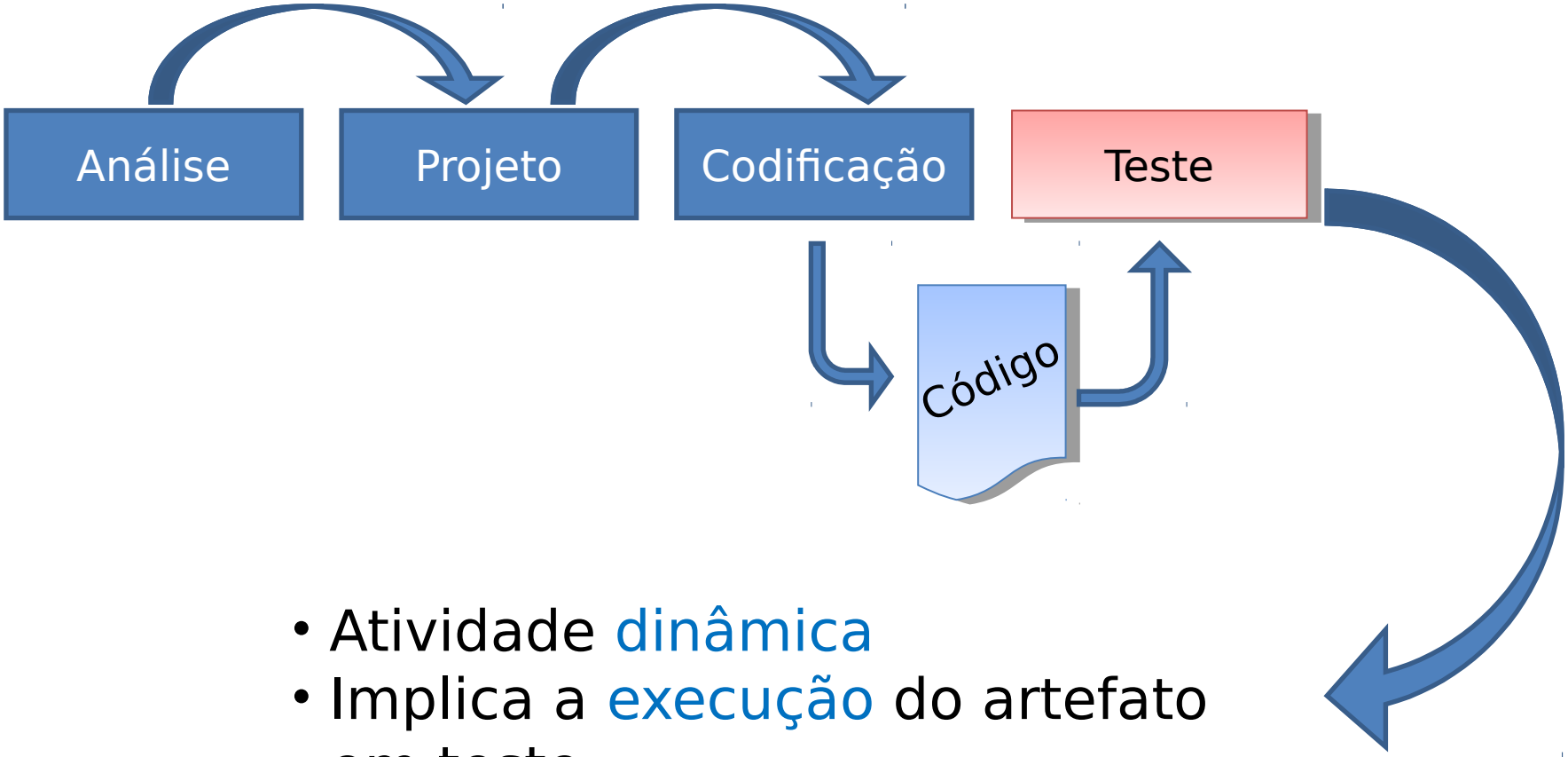


Teste de Software: Objetivos

Teste – você sabe responder ?

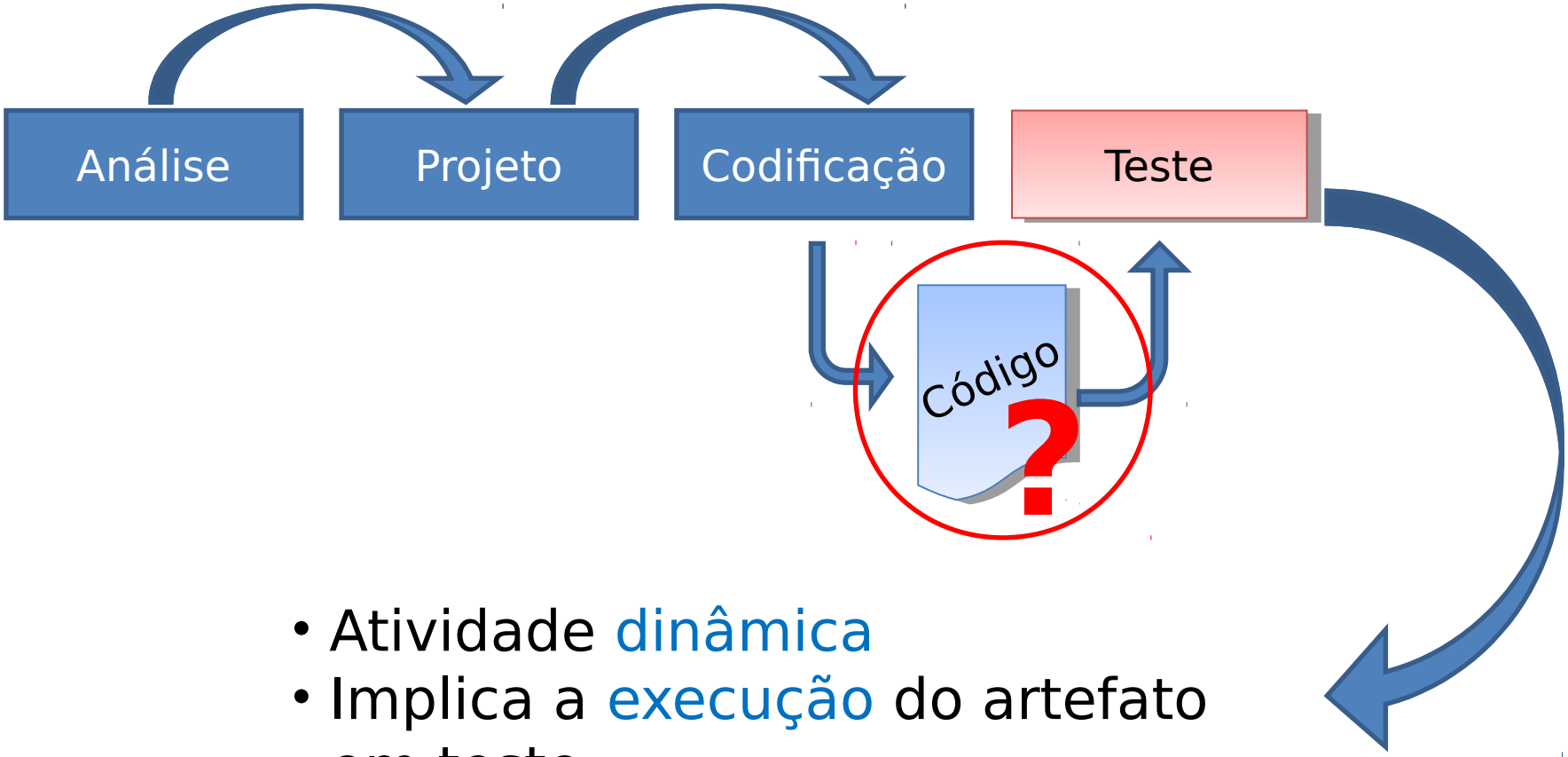
- Quando o teste ocorre?
- Qual o artefato necessário para que o teste possa existir?
- Qual o objetivo da atividade de teste?
- Depois de realizado o teste, é possível afirmar que o programa está correto?
- Quando é possível afirmar que o programa está correto?
- Quando parar a atividade de teste?

Quando o teste ocorre?



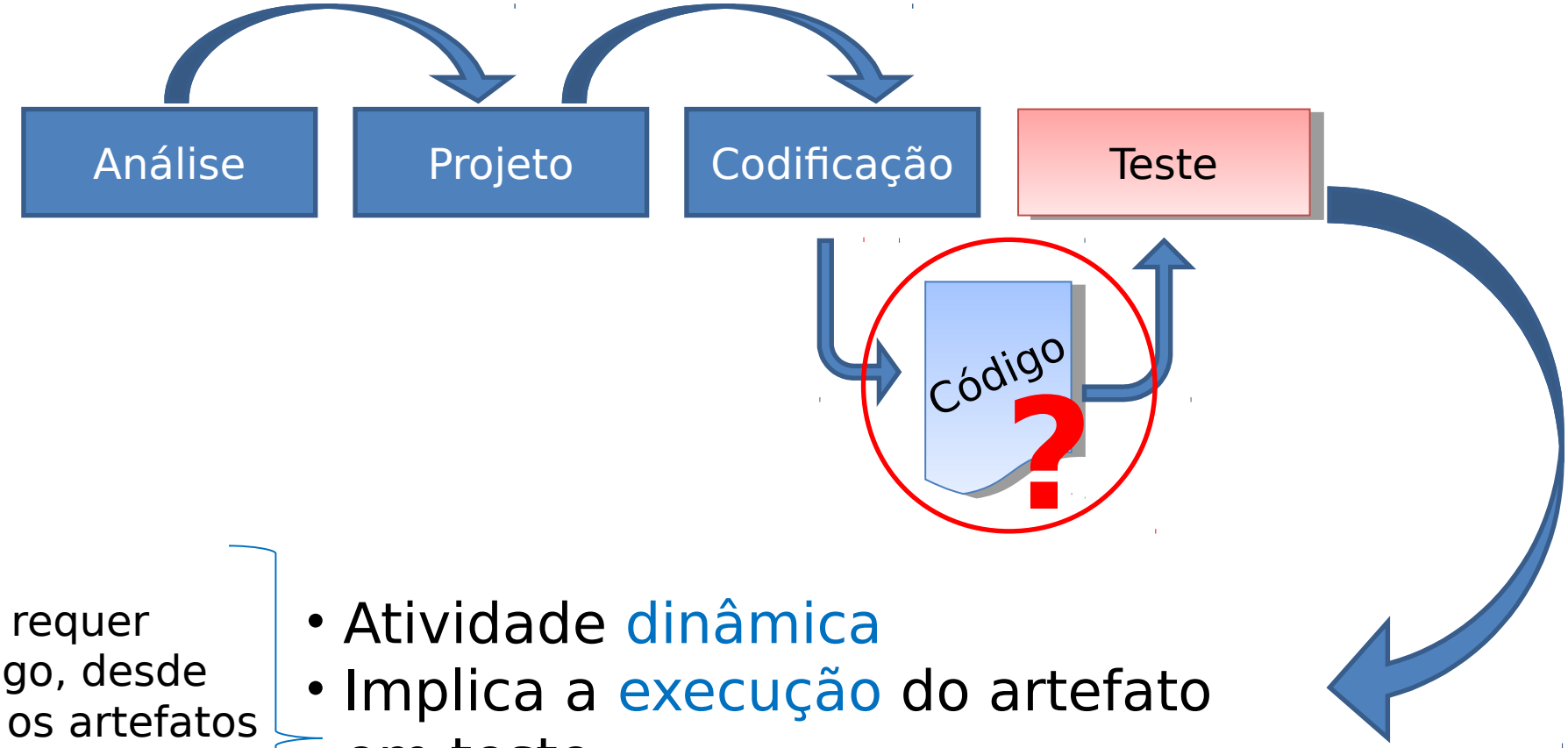
- Atividade dinâmica
- Implica a execução do artefato em teste
- Avalia o comportamento

Qual o artefato necessário para que o teste possa ocorrer?



- Atividade dinâmica
- Implica a execução do artefato em teste
- Avalia o comportamento

Qual o artefato necessário para que o teste possa ocorrer?



Não requer código, desde que os artefatos disponíveis tenham essas características...

- Atividade dinâmica
- Implica a execução do artefato em teste
- Avalia o comportamento

A Psicologia do teste de software

- Teste de software é um **processo destrutivo**, sob o ponto de vista psicológico, contrariamente às demais fases da Engenharia de Software, nas quais se constroi um produto.
- Equipes de teste são as mais “indigestas” dentre as diversas equipes que atuam em projetos de software.

A Psicologia do teste de software (Myers et al., 2004)

- Uma das principais causas de uma atividade de teste pobre é a falsa impressão de que:
  - "*Teste é o processo de demonstrar que defeitos não estão presentes.*"
  - "*O objetivo do teste de software é mostrar que o programa se comporta corretamente.*"
  - "*Teste é o processo de estabelecer uma confiança que o programa faz o que é esperado que ele faça.*"

Teste é o processo de executar um programa com o objetivo de revelar defeitos.

Myers, G. J.; Sandler, C.; Badgett, T.; Thomas, T. M. The art of software testing. 2nd ed. Hoboken/NJ - USA: John Wiley & Sons, 2004.

A Psicologia do teste de software (Myers et al., 2004)

Teste é o processo de executar um programa com o objetivo de revelar defeitos.

- Uma definição como essa pode trazer uma caráter negativo ao teste:
  - A maioria das pessoas tende a atingir objetivos, e não impedí-los de serem atingidos. No caso do software, por quê visaríamos a mostrar que algo feito por outra pessoa, ou por nós mesmos, está errado?
- Entretanto...
  - A melhor forma de mostrar a um desenvolvedor que o programa dele é “perfeito” (ou seja, livre de defeitos) é tentando, de todas as formas, refutar essa hipótese.

Qual o objetivo da atividade de teste?

É uma atividade para mostrar que o programa está correto?

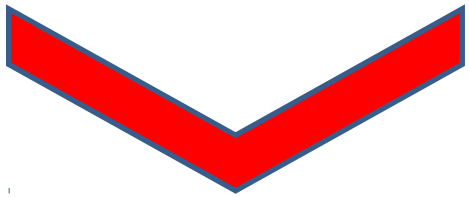
( ) SIM ( ) NÃO

Qual o objetivo da atividade de teste?

É uma atividade para mostrar que o programa está correto?

( ) SIM

(**X**) NÃO



Mas então, qual é o **objetivo** da atividade de teste?

Qual o objetivo da ativid e de teste?

É un

to?

AO

M jetivo da atividade

de teste?

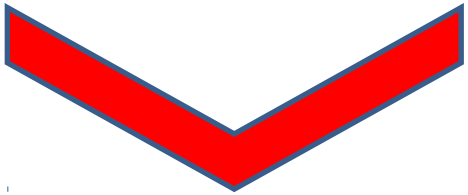
Refutar a afirmação de que
o programa está correto, isto é,
mostrar que o programa
está INCORRETO!

Depois de realizado o teste é possível afirmar que o programa está correto?

( ) SIM ( ) NÃO

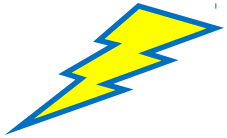
Depois de realizado o teste é possível afirmar que o programa está correto?

( ) SIM (**X**) NÃO



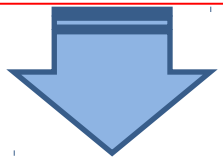
É possível aumentar a confiança de que isso seja verdade, mas não garantir! ....a não ser que se aplique o TESTE EXAUSTIVO

Quando é possível afirmar que o programa está correto?

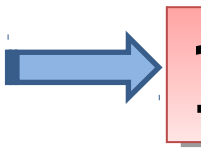


Quando se aplica TESTE EXAUSTIVO = testar com TODAS as possíveis entradas

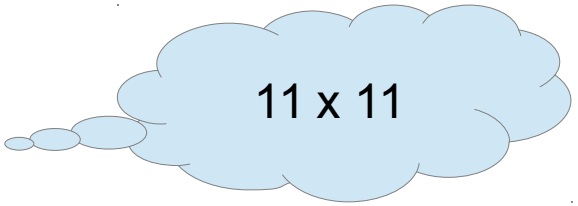
Ex: Suponha um programa que receba 2 números inteiros entre 0 e 10 e calcule a média aritmética



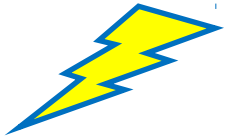
Se for aplicado o teste exaustivo, quantas são as entradas?



121

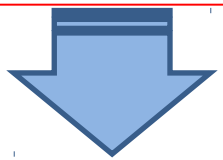


Quando é possível afirmar que o programa está correto?



Quando se aplica TESTE EXAUSTIVO = testar com TODAS as possíveis entradas

Ex: Suponha um programa que receba 2 números **reais** entre 0,0 e 10,0 e calcule a média aritmética



Se for aplicado o teste exaustivo, quantas são as entradas?

... e se forem duas casas decimais???

Quando é possível afirmar que o programa está correto?

Resumindo ……

Em geral, NÃO é possível aplicar o TESTE EXAUSTIVO

NÃO é possível afirmar que o programa está correto.

Quando parar a atividade de teste?

Terminou o tempo

Terminou o recurso financeiro

O **plano de teste** foi cumprido

O Plano de Teste

- Documento que contém todas as informações sobre a atividade de teste pretendida:
  - Equipe de teste
  - Software de suporte
  - Cronograma das atividades
  - Técnicas e critérios a serem seguidos
  - Forma de execução dos casos de teste
  - Forma de avaliação dos resultados
  - ……

Teste de Software: Conceitos Básicos

Defeito, Erro e Falha

- Defeito  Erro  Falha
  - **Defeito**: deficiência algorítmica que, se ativada, pode levar a uma falha
  - **Erro**: estado inconsistente, decorrente da execução de um defeito
  - **Falha**: evento observável que mostra que o programa violou suas especificações.
    - Ocorre quando um erro extrapola as barreiras do sistema.

Caso de Teste

- Tupla ***[d, O(d)]***, na qual:
  - ***d***: representa um elemento de um domínio ***D*** que servirá
  - ***O(d)***: saída esperada do software quando executado com a entrada **d**
- Informalmente, um caso de teste é formado por um **conjunto de entradas** e um **conjunto de saídas esperadas** após a execução do software com tais entradas.
- **Oráculo**: é quem decide se a saída obtida é de fato a saída esperada.
  - Pode ser automatizado, porém é difícil em diversas situações

Caso de Teste

- Projeto de casos de teste pode ser tão difícil quanto o projeto do próprio produto a ser testado
- Poucos programadores/analistas gostam de teste; menos ainda de projeto de casos de teste.

Caso de Teste

- Se ***D*** é muito grande ou infinito, como podemos evitar a construção de um conjunto muito grande de casos de teste?
  - **Adotando técnicas e critérios de seleção de casos de teste**

Teste de Software: Técnicas e Critérios

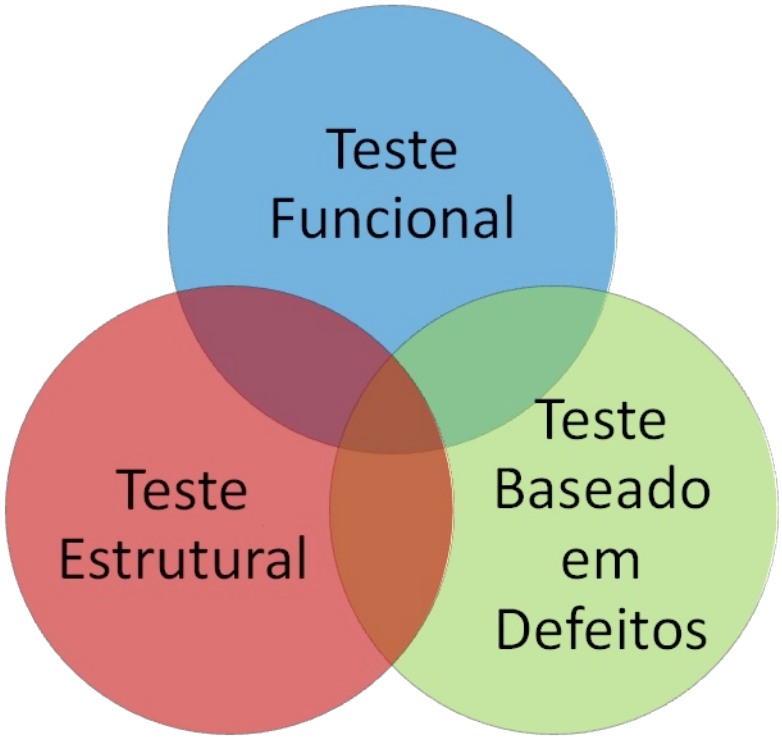
Técnicas de teste

- Formas de se testar o software baseadas em diferentes artefatos de software disponíveis.
- Relacionadas aos diferentes “pontos de vista” adotados para se testar um software:
  - Ponto de vista do especificação funcional (ou não funcional) do software
    - **Teste funcional**
  - Ponto de vista da estrutura interna
    - **Teste estrutural**
  - Ponto de vista dos defeitos mais recorrentes
    - **Teste baseado em defeitos**

Técnicas de teste

Teste funcional	Os requisitos de teste são extraídos da especificação do software
Teste estrutural	Os requisitos de teste são extraídos da estrutura do software, em variados níveis de granularidade
Teste baseado em defeitos	Os requisitos de teste são extraídos dos defeitos típicos e comuns inseridos durante a construção do software

Técnicas de teste

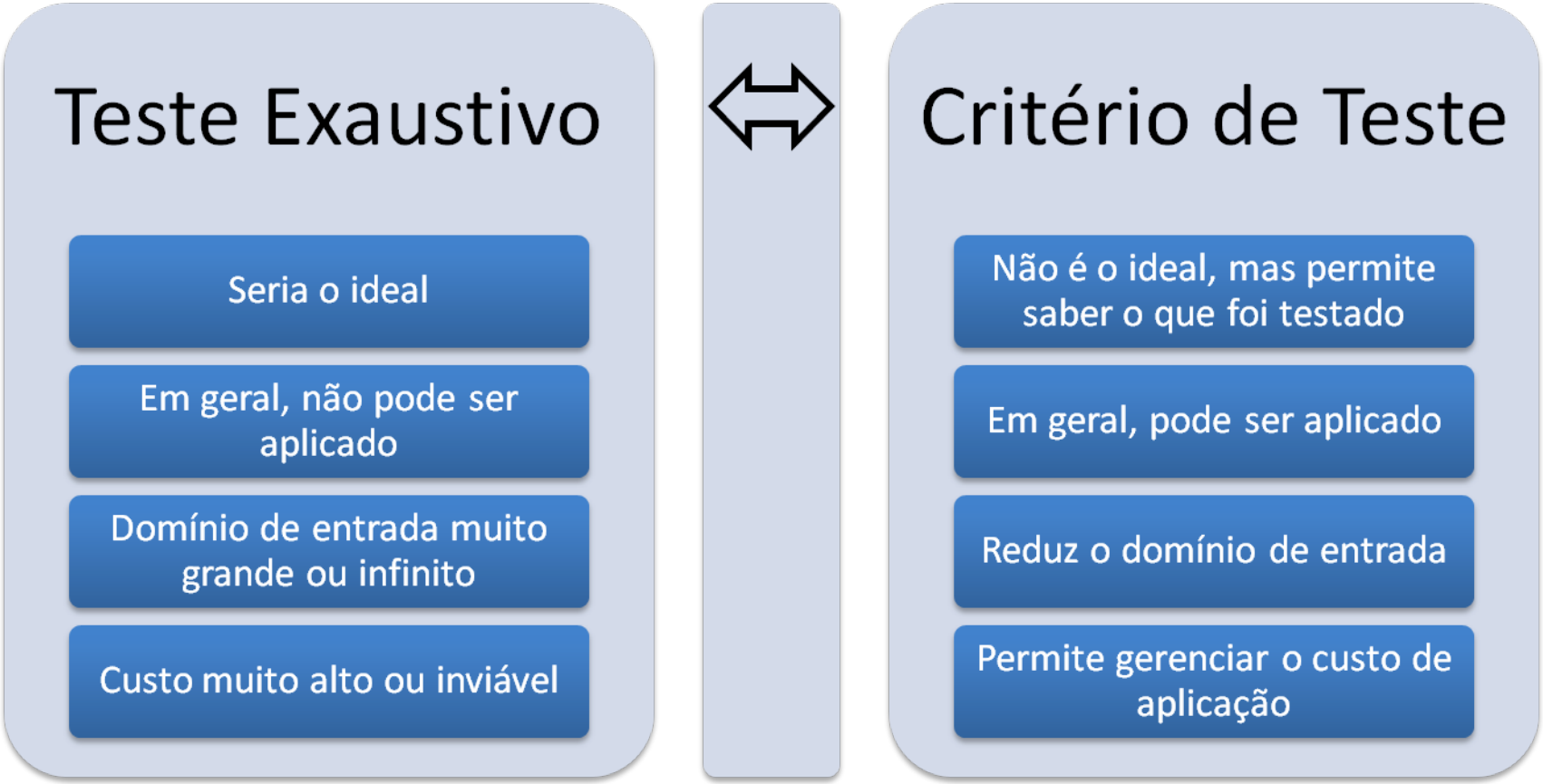


São complementares, pois identificam tipos diferentes de defeitos.

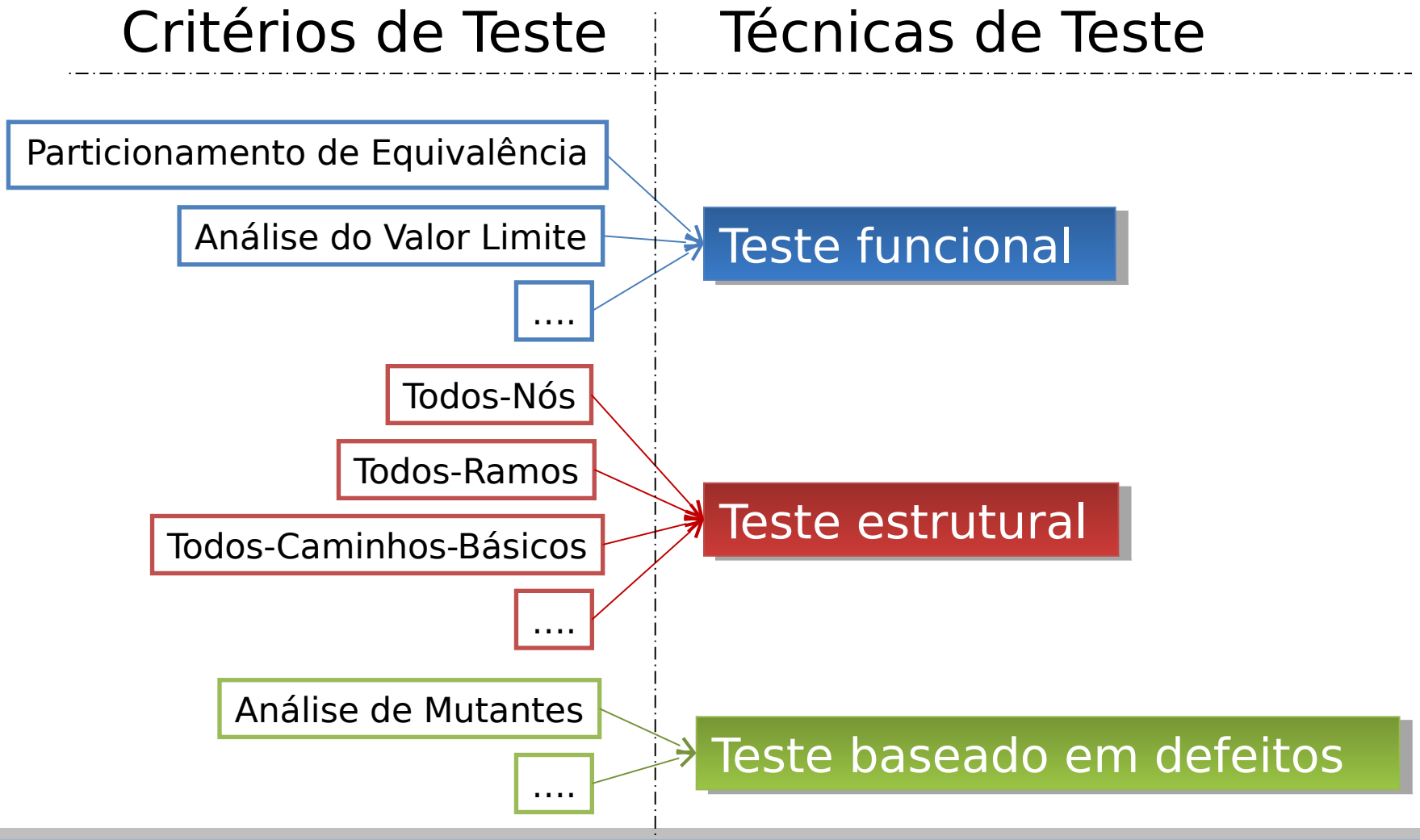
Critérios de Teste: O que são e porque são necessários?

- Maneira sistemática e planejada de elaborar os casos de teste (CT)
- Podem ser usados de duas formas:
  - Para seleção dos casos de teste: quando os CTs são criados para satisfazer os requisitos do critério de teste
  - Para adequação dos casos de teste: quando os CTs são criados e.g. aleatoriamente, e então se verifica se eles atendem (ou satifazem) os requisitos do critério de teste.

Critérios de Teste: O que são e porque são necessários?



Critérios de Teste & Técnicas de Teste



Sumário

Pontos-Chaves

- O objetivo do teste é revelar detectar e se eles não forem detectados, o teste não pode afirmar sua ausência.
- Testar tudo é impossível!!!
- As técnicas de teste são complementares, isto é, devem aplicadas conjuntamente.
- A execução do teste é criativa e difícil, pois para testar com eficiência é preciso conhecer o software a fundo.