

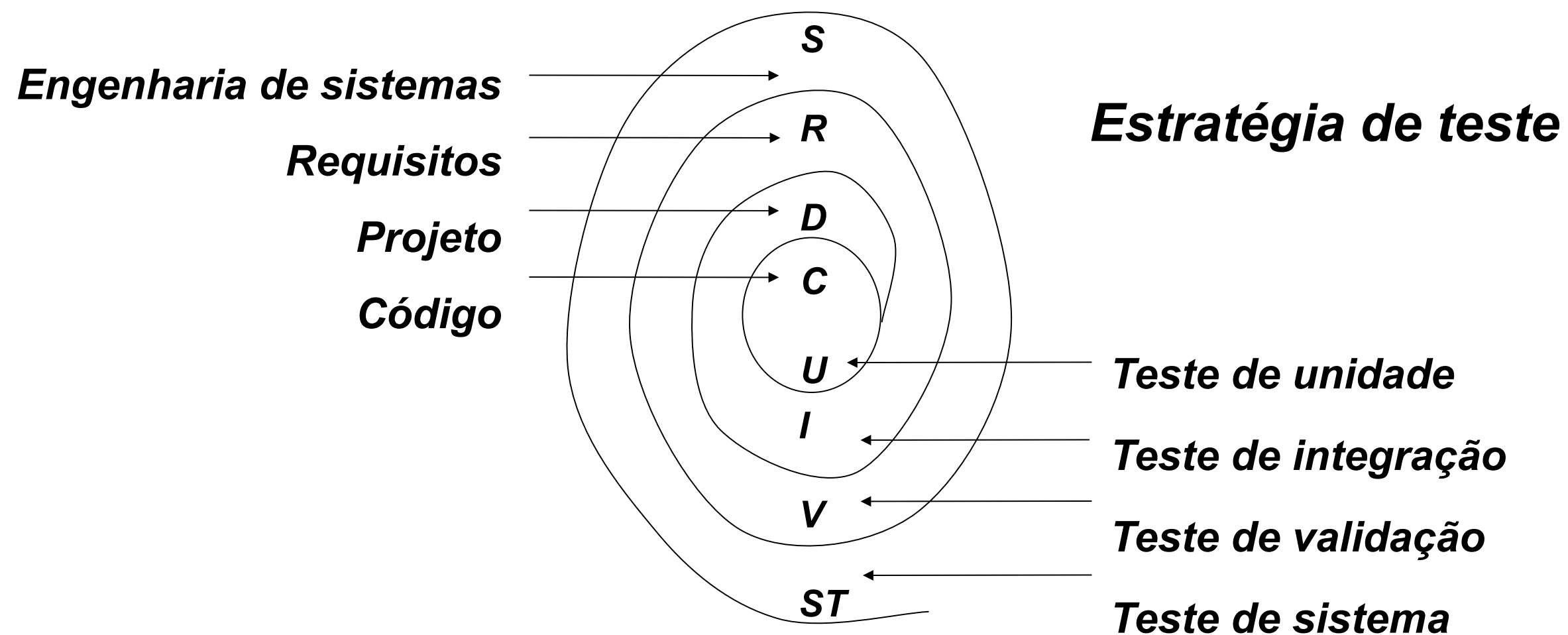
# ***Plano de Testes***

# ***Introdução***

***Teste é um conjunto de atividades que pode ser planejado antecipadamente e realizado sistematicamente.***

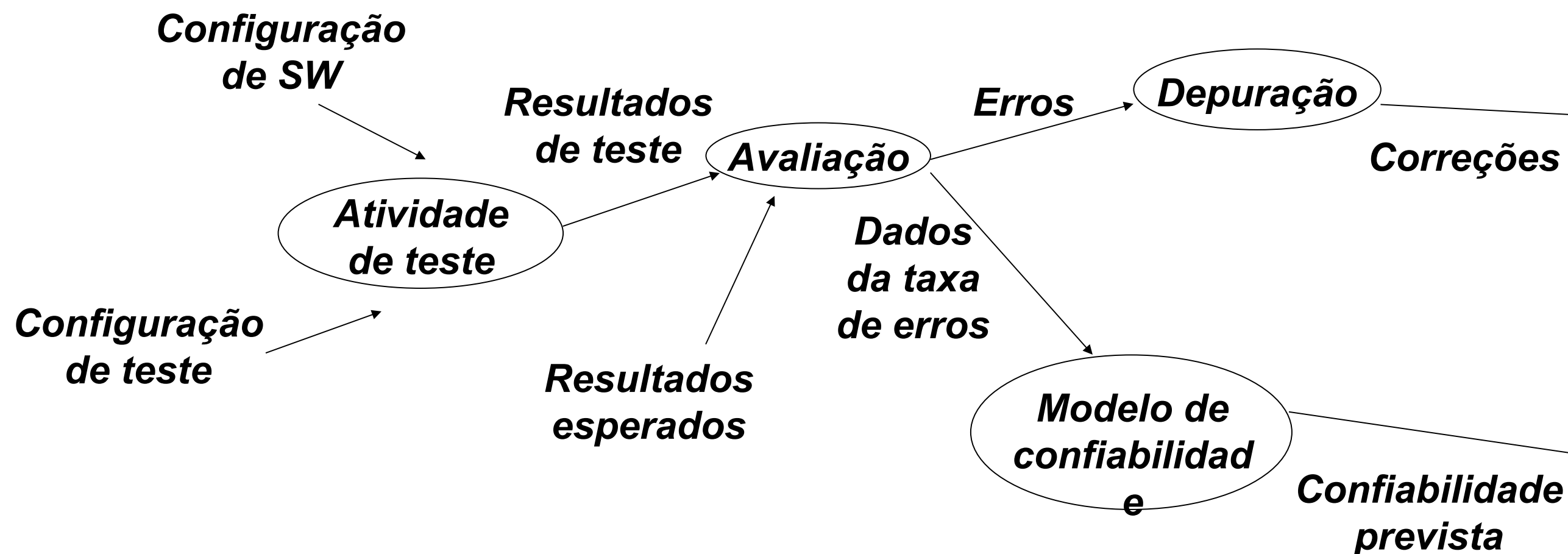
***É possível definir um “template” (esqueleto), ou seja um conjunto de passos ao qual é possível alocar técnicas de projeto de casos de teste e estratégias de teste específicos.***

# Objetivos do Teste



**O Processo de Teste, como qualquer outro processo deve ser revisto continuamente, de forma a ampliar sua atuação e possibilitar aos profissionais uma maior visibilidade e organização dos seus trabalhos, o que resulta numa maior agilidade e controle operacional dos projetos de testes.**

# *Fluxo de informações de teste*



- **O processo de depuração é a parte mais imprevisível do processo de teste. Um erro que indique uma discrepância de 0,01% entre resultados esperados e reais pode demorar uma hora, um dia ou um mês para ser diagnosticado e corrigido.**

# *Teste de Caixa Preta*

- ***Teste de caixa preta refere-se aos testes realizados nas interfaces do SW (a entrada é adequadamente aceita e a saída é corretamente produzida com a integridade das informações externas mantida).***

# *Teste de Caixa Branca*

***Teste de caixa branca baseia-se num minucioso exame dos detalhes procedimentais, através da definição de todos os caminhos lógicos possíveis.***

***Infelizmente estes testes apresentam problemas logísticos, uma vez que o número destes possíveis caminhos lógicos pode ser muito grande, o que levaria a um tempo infinito.***

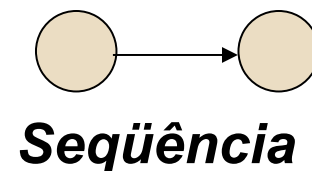
***Entretanto este tipo de teste não pode ser desprezado como pouco prático, podendo-se optar por um número limitado de opções***

# Teste de caminho básico

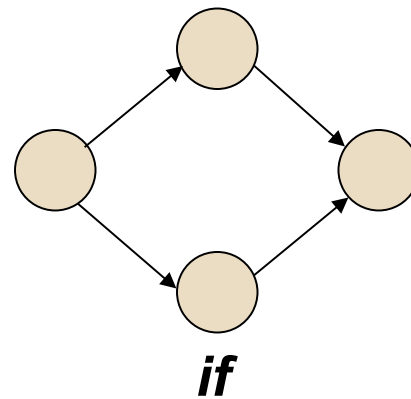
***Técnica de teste de caixa branca que possibilita que o projetista do caso de teste derive uma medida de complexidade lógica de um projeto procedimental e use essa medida como guia para definir um conjunto básico de caminhos de execução.***

## **Notação de grafo de fluxo:**

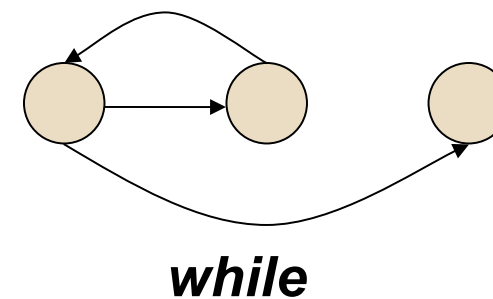
- notação simples para representação do fluxo de controle, que descreve o fluxo lógico:***



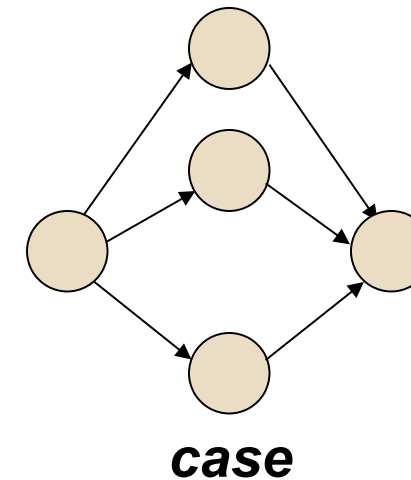
***Seqüência***



***if***



***while***



***case***



# ***Complexidade Ciclomática***

***É uma métrica de SW que proporciona uma medida quantitativa da complexidade lógica de um programa***

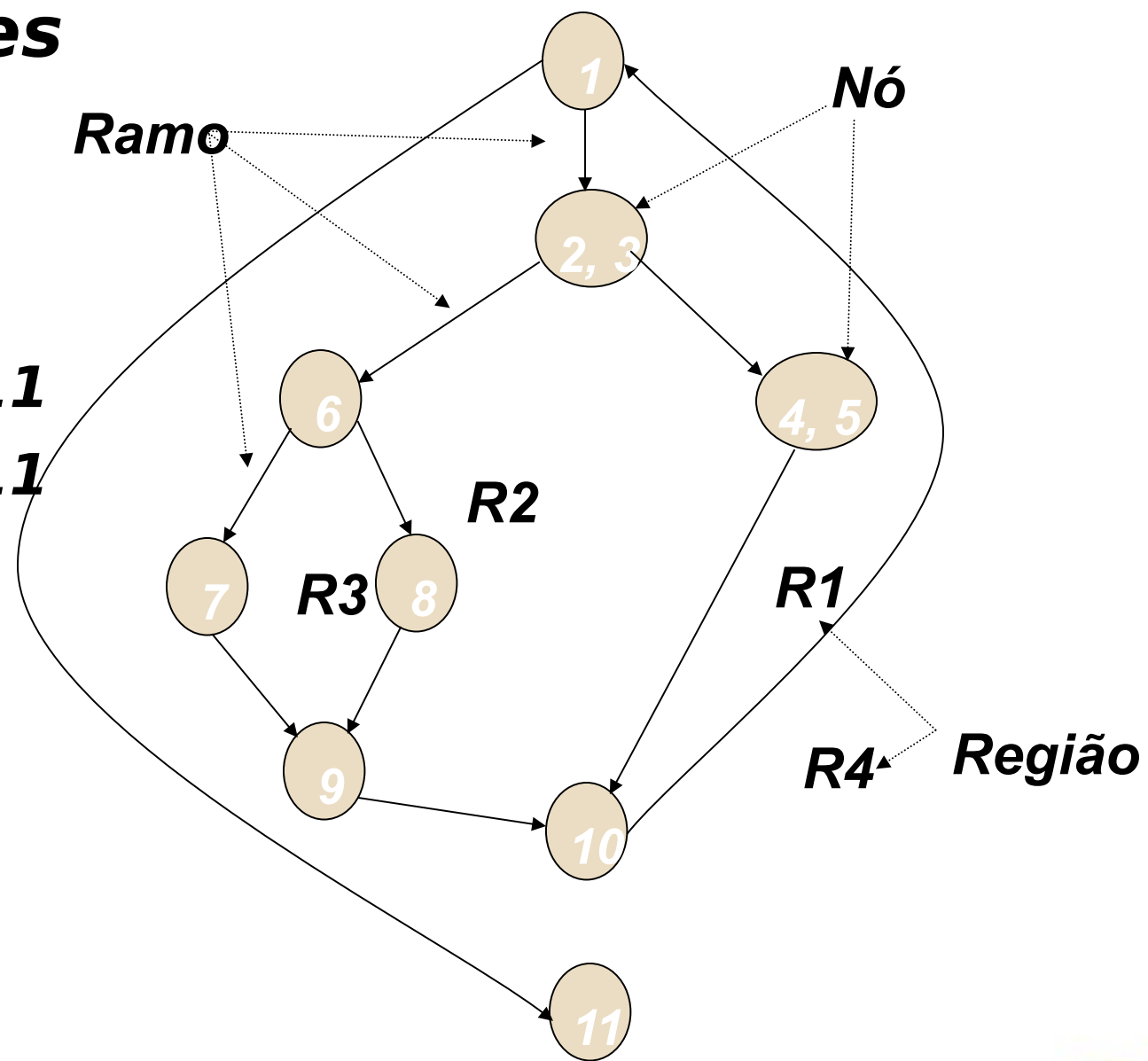
***O valor computado da complexidade ciclomática define o número de caminhos independentes do conjunto básico de um programa e oferece-nos um limite máximo para o número de testes que deve ser realizado para garantir que todas as instruções sejam executadas pelo menos uma vez.***



# Complexidade Ciclomática

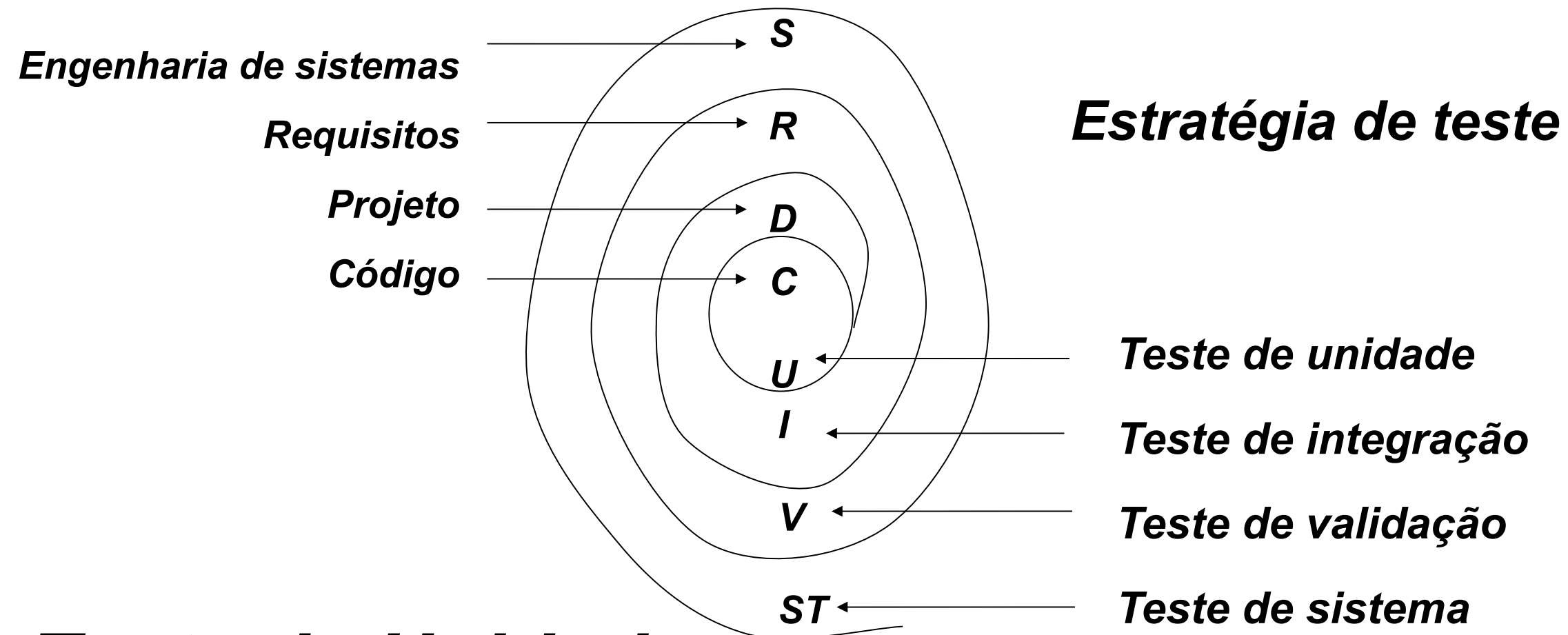
- ***Por exemplo, um conjunto de caminhos independentes, referentes à figura ao lado:***

- ***caminho 1: 1-11***
- ***caminho 2: 1-2-3-4-5-10-1-11***
- ***caminho 3: 1-2-3-6-8-9-10-1-11***
- ***caminho 4: 1-2-3-6-7-9-10-1-11***



## Grafo de fluxo

# *Estratégias de Teste de Software*



- **Teste de Unidade**
- **Teste de Integração**
- **Teste de Validação**
- **Teste de Sistema**

# *Testes de Unidade*

***Concentra-se no esforço de verificação da menor unidade de projeto de SW - o módulo. Baseia-se quase sempre na técnica de caixa branca (com menor incidência na O.O.) e pode ser realizado em paralelo para múltiplos módulos.***

# *Testes de Integração*

***O objetivo é, a partir dos módulos testados no nível de unidade, construir a estrutura de programa que foi determinada pelo projeto realizando-se ao mesmo tempo, testes para descobrir erros associados a interfaces (entradas e saídas entre módulos devem se compatibilizar).***

# *Testes de Validação*

- *São definidas expectativas razoáveis na Especificação de Requisitos de SW, que descreve todos os atributos do SW visíveis ao usuário.*
- *A validação é bem-sucedida quando o SW funciona de uma maneira razoavelmente esperada pelo cliente.*

# *Testes de Sistema*

***É uma série de diferentes testes, cujo propósito primordial é pôr completamente à prova o sistema baseado em computador.***

# Teste de Sistema

- **Teste de recuperação:** é um teste de sistema que força o SW a falhar de diversas maneiras e verifica se a recuperação é adequadamente executada.
- **Teste de segurança:** tenta verificar se todos os mecanismos de proteção embutidos em um sistema o protegerão, de fato, de acessos indevidos.
- **Teste de estresse:** executa o sistema de uma forma que exige recursos em quantidade. Essencialmente o analista tenta destruir o programa.
- **Teste de desempenho:** é idealizado para testar o desempenho de “runtime” do SW dentro do contexto de um sistema integrado.



# *Frameworks para Testes de Unidade*

***Similares ao JUnit (linguagem Java):***

***Python***

***PyUnit***

***C++***

***CppUnit***

***Perl***

***PerlUnit***

***.NET***

***NUnit, NUnitForms, dotUnit, EasyMock.NET, csUnit***

# *Planejamento de Testes*

- ***Definição de uma proposta de testes baseada nas expectativas do Cliente em relação à :***
  - *prazos,*
  - *custos*
  - *qualidade esperada*
  
- ***Possibilidade de dimensionar a equipe e estabelecer um esforço de acordo com as necessidades apontadas pelo Cliente.***

# Especificação dos Testes

- **Identificação dos casos de testes que deverão ser construídos e/ou modificados em função das mudanças solicitadas pelo Cliente.**

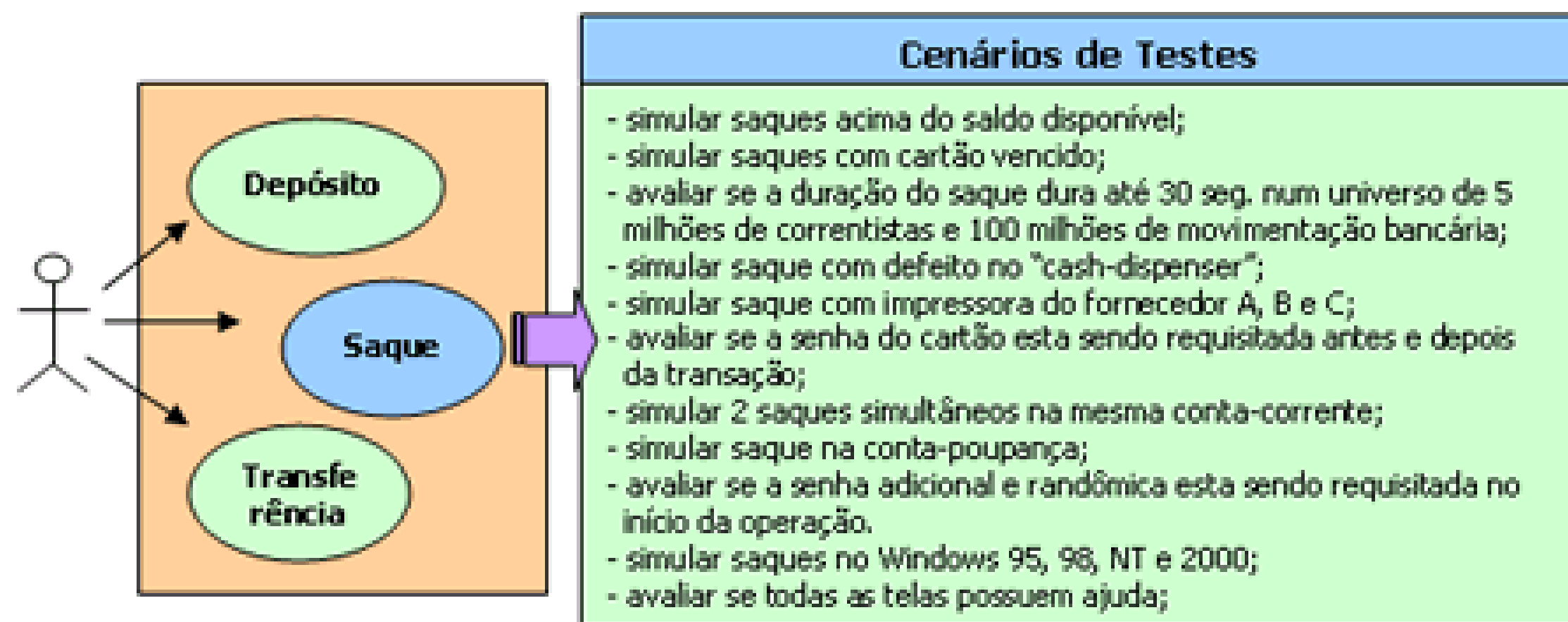


Figura - Levantamento dos Cenários sem aplicar os conceitos de Categorização

# Especificação dos Testes (Categorias)

Funcional	Segurança	Usabilidade	Performance
<ul style="list-style-type: none"> <li>- simular saques acima do saldo disponível;</li> <li>- simular saque na conta-poupança;</li> <li>- <b>simular saque acima do valor do limite da conta;</b></li> <li>- <b>simular saque com valores não múltiplos das notas;</b></li> <li>- <b>simular saque com valores não múltiplos das notas;</b></li> </ul>	<ul style="list-style-type: none"> <li>- simular saques com cartão vencido;</li> <li>- avaliar se a senha do cartão esta sendo requisitada antes e depois da transação;</li> <li>- avaliar se a senha adicional e randômica esta sendo requisitada no início da operação;</li> <li>- <b>simular saque noturno acima do valor permitido;</b></li> </ul>	<ul style="list-style-type: none"> <li>- avaliar se todas as telas possuem ajuda;</li> <li>- <b>avaliar se mensagens são claras e objetivas;</b></li> <li>- <b>avaliar se o padrão visual é mantido em todos os momentos;</b></li> <li>- <b>avaliar se todas as operações possuem caminhos de fuga;</b></li> </ul>	<ul style="list-style-type: none"> <li>- avaliar se a duração do saque dura até 30 seg. num universo de 5 milhões de correntistas e 100 milhões de movimentação bancária;</li> <li>- <b>garantir que manipulação com dispositivos físicos no saque não ultrapassem 10 seg. da operação;</b></li> </ul>
Carga e Concorrência	Configuração	Recuperação	Contingência
<ul style="list-style-type: none"> <li>- simular 2 saques simultâneos na mesma conta-corrente;</li> <li>- <b>simular 10.000 saques simultâneos;</b></li> </ul>	<ul style="list-style-type: none"> <li>- simular saque com impressora do fornecedor A, B e C;</li> <li>- simular saques no Windows 95, 98, NT e 2000;</li> <li>- <b>simular saque com impressora do fornecedor X, Y e Z;</b></li> </ul>	<ul style="list-style-type: none"> <li>- simular saque com defeito no "cash-dispenser";</li> <li>- <b>simular saque com defeito na impressora;</b></li> <li>- <b>simular saque com falha de conexão com a central;</b></li> <li>- <b>simular saque com queda de energia;</b></li> </ul>	<ul style="list-style-type: none"> <li>- <b>disparar processo de instalação emergencial;</b></li> </ul>

# ***Modelagem dos Testes***

- ***Identificação de todos os elementos necessários para a implementação de cada caso de teste especificado:***
  - ***modelagem das massas de testes***
  - ***definição dos critérios de tratamento de arquivos (descaracterização e comparação de resultados).***

# *Preparação do Ambiente*

- ***Conjunto de atividades que visa a disponibilização física de um ambiente de testes para sofrer a bateria de testes planejadas nas etapas anteriores de forma contínua e automatizada (sem intervenção humana).***

# *Execução dos Testes*

- ***Execução e conferência dos testes planejados, de forma a garantir que o comportamento do aplicativo permanece em "conformidade" com os requisitos contratados pelo Cliente.***



# *Análise dos Resultados*

- ***Análise e confirmação dos resultados relatados durante a fase de execução dos testes.***
- ***Os resultados em "não-conformidade" deverão ser "confirmados" e "detalhados" para que a Fábrica de Software realize as correções necessárias.***
- ***Já os em "conformidade" deverão ter seu resultado "POSITIVO" reconfirmado.***

## *Norma IEEE 829-1998*

***A norma IEEE 829-1998 descreve um conjunto de documentos para as atividades de teste de um produto de software. Os documentos cobrem as tarefas de planejamento, especificação e relato de testes.***