

Teste Funcional ou Caixa-Preta

Objetivos

- Apresentar a técnica de teste funcional ou caixa-preta
- Mostrar exemplos de casos de teste criados para satisfazer os principais critérios da técnica de teste funcional
- Propor exercícios para exercitar a técnica de teste funcional

Conceito de Teste de Software

- Consiste em executar um programa com o objetivo de revelar uma falha (Myers, 1979)
- Uma falha é qualquer evento do sistema que viola um objetivo de qualidade estabelecido
- Exemplos típicos de falhas (failures)
 - Crash (erros de runtime)
 - Resultado errado
 - O usuário não consegue executar uma operação do software
 - Tempo de resposta excedido
 - Formato de saída fora do padrão
 - A senha digitada não aparece ofuscada
 - O link está “quebrado”

Caso de teste típico

- Configuração/estado do sistema antes da execução do teste, pré-condições
- Ações, operações, tarefas a serem executadas
- Resultado
- Mensagem de erro
- Página/tela alcançada
- Prioridade
- Status
- Informações sobre o ambiente de execução
- Comentários
- ID, nome
- Cenário de teste
- Descrição do caso de teste
- Os dados de teste
- Os resultados esperados (oráculos)

Exemplo

- ID:
 - 01
- Nome:
 - Login bem sucedido
- Descrição:
 - Usuário acessa o aplicativo do BB
 - Usuário aperta o botão “Entrar”
 - Usuário fornece dados de login (agência, conta corrente e senha)
 - Usuário aperta o botão “Entrar”
- Pré-condição:
 - Usuário possui conta no banco e senha para mobile banking cadastrada
- Dados de teste:
 - 1234, 12323-4, 24781643

Exemplo

- Resultado esperado:
 - Login bem sucedido
 - Abrir tela com operações disponíveis
- Prioridade:
 - Alta
- Testador:
 - Eler
- Status
 - Não-executado

Exemplo - conjunto de casos de teste

I D	Nom e	Descrição	Pré- condição	Dados	Resultado esperado	Prioridad e	Testado r	Status
1								
2								
.. .								
N								

Execução dos casos de teste

- Manual
- Automatizada
 - Scripts de teste
 - Frameworks de teste
 - Record/Play

Como encontrar todas as falhas de um software?

- Um conjunto de casos de teste com **qualidade ótima** é capaz de revelar todas as falhas de um software
- Para encontrar todas as falhas e garantir que o software está livre de defeitos é necessário **testá-lo em todas as condições** e com **todos os dados de entrada possíveis**

Como encontrar todas as falhas de um software?

- Entretanto, as condições, os cenários, os dados de entrada e suas combinações tendem a ser **infinitos** ou **muito grandes**
- Portanto, criar um conjunto de casos de teste ótimo é geralmente **impossível** ou **impraticável**

Uma pergunta importante neste cenário

- Como criar um conjunto de casos de teste que seja:
 - Finito e factível?
 - Capaz de revelar o maior número de falhas perceptíveis?
 - Capaz de revelar as falhas mais críticas ou relevantes do software?
- Esta pergunta esconde algumas outras:
 - Como seleccionar cenários/dados para criar bons casos de teste?
 - Quantos testes são necessários para testar um software ou parte dele?
 - Como saber se os testes criados são suficientes?

Exemplo: IRPF 2021 (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	

IRPF 2021 (uma simplificação) - CT1

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	9000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2
Total a pagar ou a receber:	

IRPF 2021 (uma simplificação) - CT2

Entrada	Valor
Rend. tributáveis recebidos PJ:	150000
Rend. isentos e não-tributáveis:	30000
Imposto pago (retido na fonte):	40000
Pagamentos efetuados (educação):	12000
Pagamentos efetuados (saúde):	15000
Número de dependentes:	3
Total a pagar ou a receber:	

IRPF 2021 (uma simplificação) - CT3

Entrada	Valor
Rend. tributáveis recebidos PJ:	200000
Rend. isentos e não-tributáveis:	30000
Imposto pago (retido na fonte):	60000
Pagamentos efetuados (educação):	15000
Pagamentos efetuados (saúde):	18000
Número de dependentes:	3
Total a pagar ou a receber:	

IRPF 2021 (uma simplificação) - CT4

Entrada	Valor
Rend. tributáveis recebidos PJ:	1000000
Rend. isentos e não-tributáveis:	30000
Imposto pago (retido na fonte):	200000
Pagamentos efetuados (educação):	35000
Pagamentos efetuados (saúde):	28000
Número de dependentes:	2
Total a pagar ou a receber:	

Perguntas

- Esses casos de teste são capazes de:
 - revelar o maior número de falhas perceptíveis?
 - revelar as falhas mais críticas ou relevantes do software?
 - são suficientes?

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	

Abstraíndo

- Imagine que você é o novo Presidente da República
- Sua primeira função é nomear os ministros ou ministras dos seguintes ministérios federais:
 - Defesa
 - Economia
 - Educação
 - Saúde
 - Infraestrutura
 - Relações Exteriores

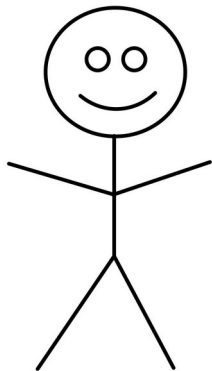
Abstraindo



DEFESA



ECONOMIA



EDUCAÇÃO



SAÚDE



REL. EXTERIORES



INFRAESTRUTURA

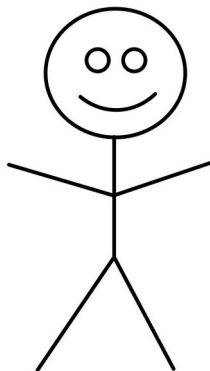
Abstraíndo



DEFESA



ECONOMIA



REL. EXTERIORES



INFRAESTRUTURA

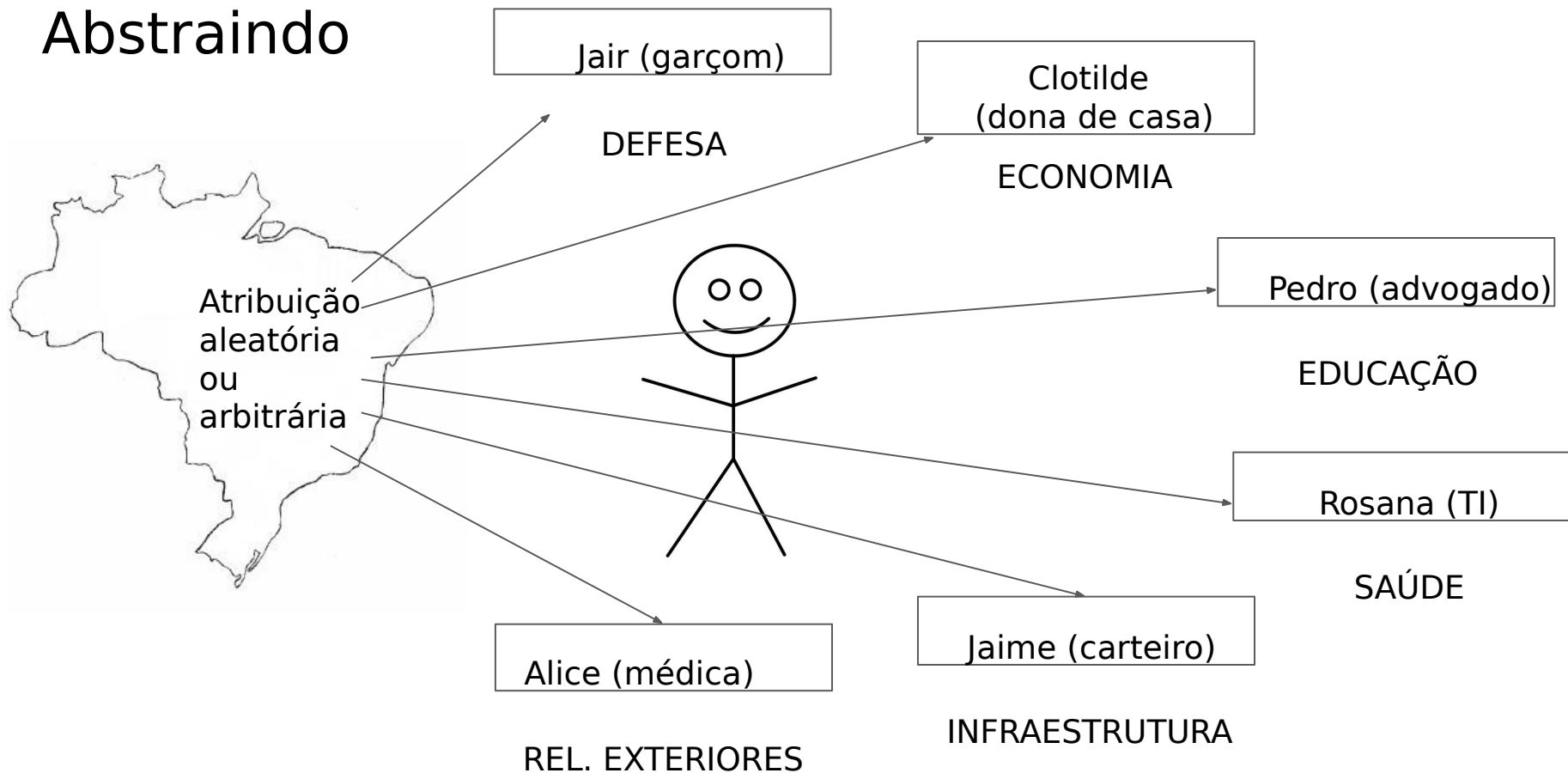


EDUCAÇÃO

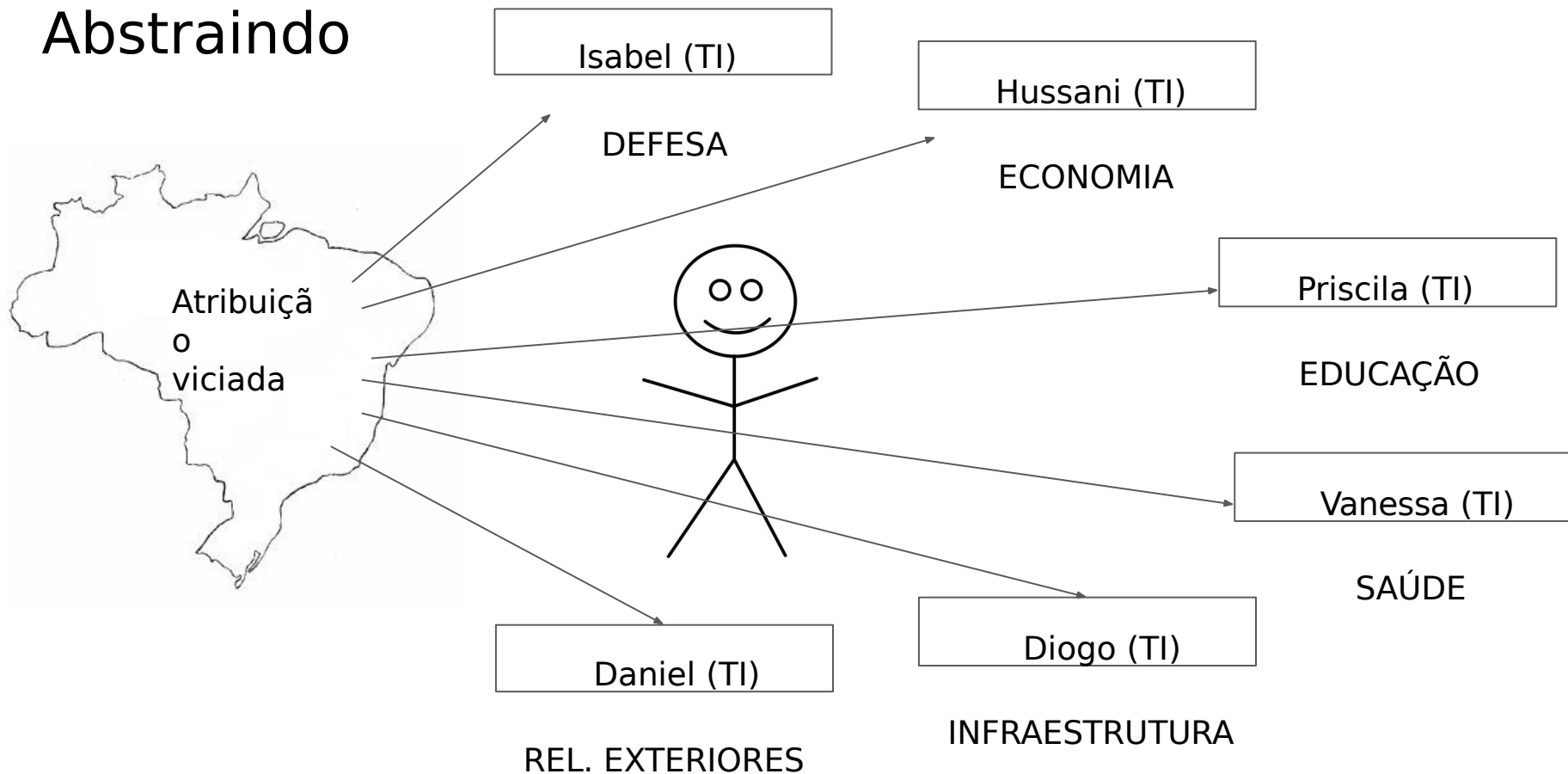


SAÚDE

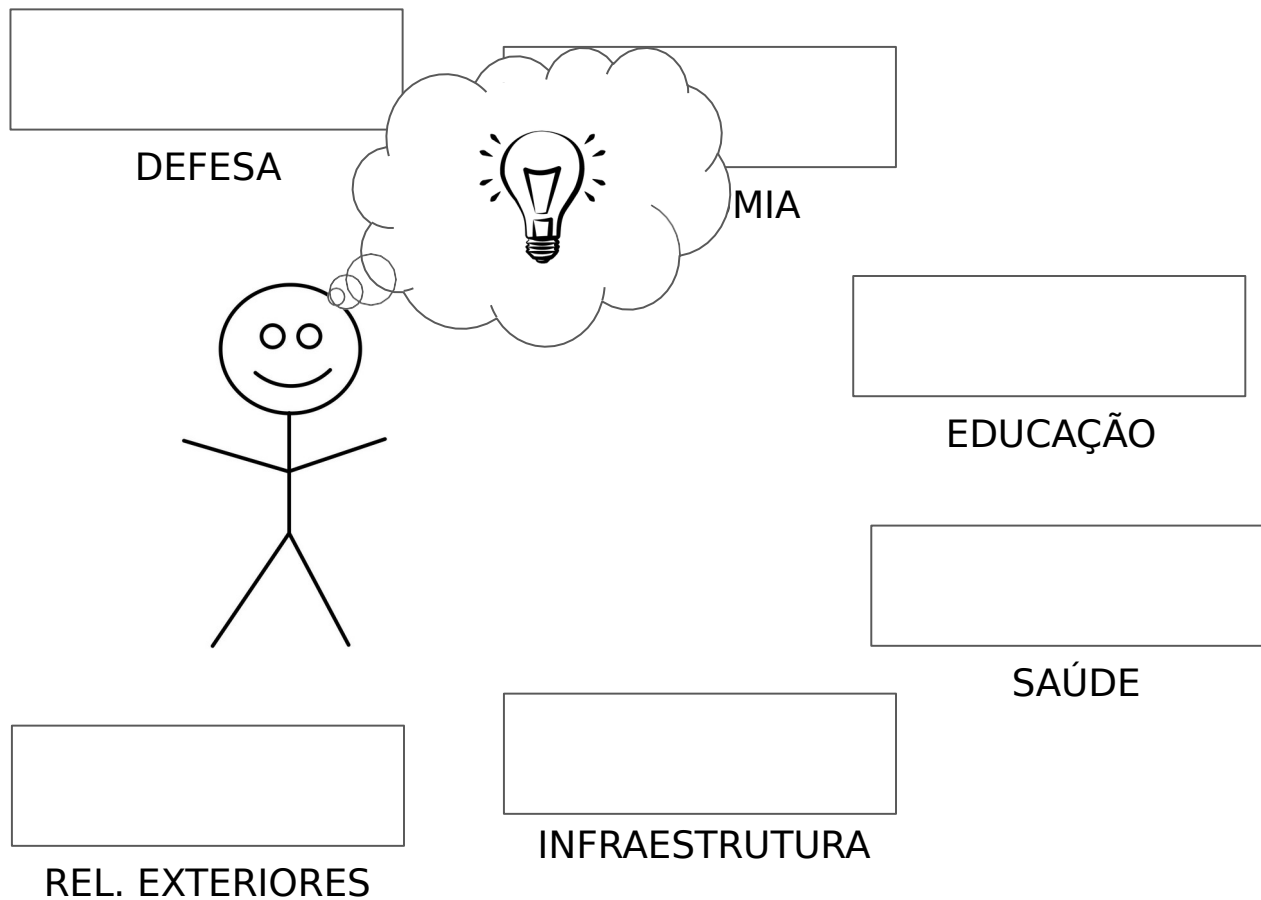
Abstraíndo



Abstraíndo



Abstraíndo



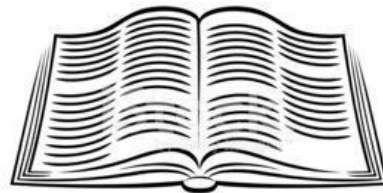
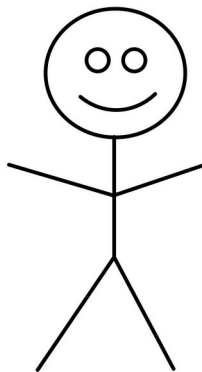
Abstraíndo



DEFESA



ECONOMIA



EDUCAÇÃO



SAÚDE



REL. EXTERIORES



INFRAESTRUTURA

Abstraíndo



EXÉRCITO
BRASILEIRO

Fábio (general)

DEFESA

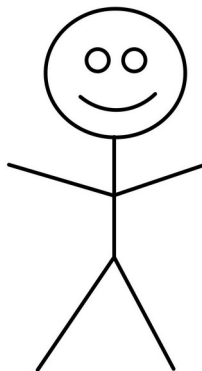
ECONOMIA

EDUCAÇÃO

SAÚDE

INFRAESTRUTURA

REL. EXTERIORES



Abstraíndo

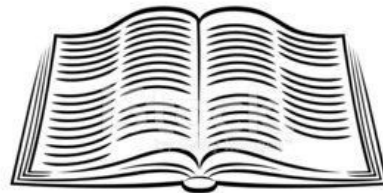
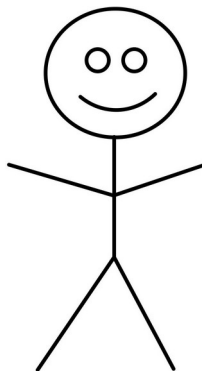


Fábio
(general)

DEFESA

Carla
(economista)

ECONOMIA



EDUCAÇÃO



SAÚDE



REL. EXTERIORES



INFRAESTRUTURA

Abstraíndo



Fábio (general)

DEFESA

Carla
(economista)

ECONOMIA

Sílvio (dir. ensino)

EDUCAÇÃO

SAÚDE

INFRAESTRUTURA

REL. EXTERIORES

Abstraíndo

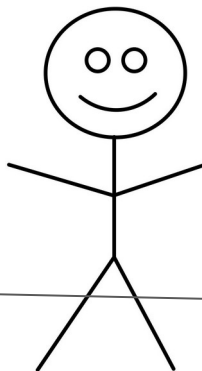


Fábio (general)

DEFESA

Carla
(economista)

ECONOMIA



Sílvio (dir. ensino)

EDUCAÇÃO

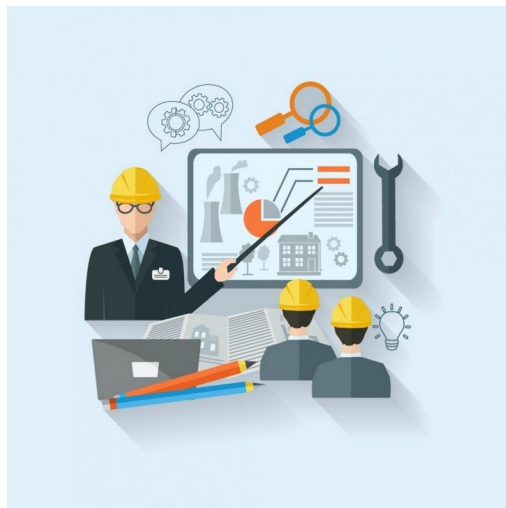
Antônio (médico)

SAÚDE

REL. EXTERIORES

INFRAESTRUTURA

Abstraíndo



Fábio (general)

DEFESA

Carla
(economista)

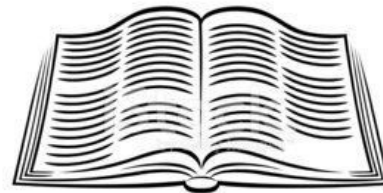
ECONOMIA

Sílvio (dir. ensino)

EDUCAÇÃO

Antônio (médico)

SAÚDE



Sandra (eng. civil)

INFRAESTRUTURA

REL. EXTERIORES

Abstraindo



Fábio (general)

DEFESA

Carla
(economista)

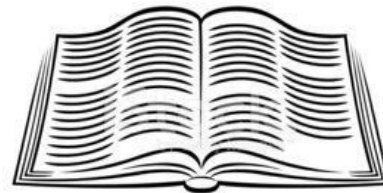
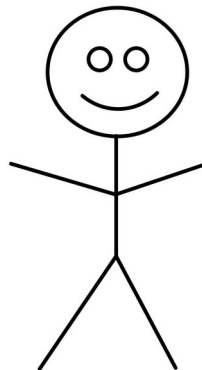
ECONOMIA

Sílvio (dir. ensino)

EDUCAÇÃO

Antônio (médico)

SAÚDE



Orlando
(advogado)

REL. EXTERIORES

Sandra (eng. civil)

INFRAESTRUTURA

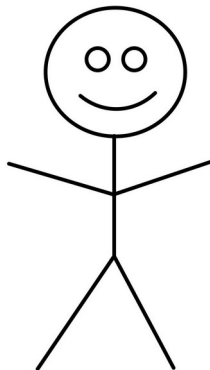
Abstraíndo



DEFESA



ECONOMIA



REL. EXTERIORES



INFRAESTRUTURA



EDUCAÇÃO



SAÚDE

Outro

exemplo...

- Técnico de futebol
- Precisa escolher:
 - Zagueiros
 - Laterais
(esquerdo/direito)
 - Meio campo
 - Atacantes
 - Goleiros
 - etc...

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	Pagamentos efetuados

(saúde): Número de dependentes:

Total a pagar ou a receber:

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados	
(educação):	

específicas?

Pagamentos efetuados (saúde): Número de dependentes:

Técnicas e critérios de teste

- Um bom caso de teste é aquele que tem uma alta probabilidade de revelar uma falha ainda não descoberta
- Portanto, técnicas e critérios de teste foram propostos para orientar o testador na tarefa de derivar os casos de teste com base nos artefatos do projeto (especificação, código, modelos, etc):
 - Teste Adhoc
 - Teste Exploratório
 - Teste Funcional ou Caixa-Preta
 - Teste Estrutural ou Caixa-Branca
 - Teste Baseado em Erros

Técnicas de teste

- Um bom caso de teste é aquele que tem uma alta probabilidade de revelar uma falha ainda não descoberta
- Portanto, técnicas de teste foram propostas para orientar o testador na tarefa de derivar os casos de teste com base nos artefatos do projeto (especificação, código, modelos, etc):
 - Teste Adhoc
 - Teste Exploratório
 - **Teste Funcional ou Caixa-Preta**
 - Teste Estrutural ou Caixa-Branca
 - Teste Baseado em Erros

Cr terios de teste

- Definem requisitos de teste que devem ser satisfeitos pelos casos de teste
- Um crit rio   satisfeito somente quando todos os requisitos que ele define s o satisfeitos
- Ajudam a responder  s quest es:
 - Como selecionar valores de entrada para criar bons casos de teste?
 - Quantos casos de teste devem ser criados?
 - Quando parar de testar?

Teste funcional ou caixa-preta

- Tem o objetivo de verificar se uma funcionalidade, operação, função, método, classe, programa está de acordo com os requisitos especificados
- Os casos de teste são definidos com base na especificação do software (descrições, casos de uso, requisitos, diagramas, etc)
- Exemplos de critérios desta técnica:
 - Classes de equivalência
 - Análise de valor-limite

Teste funcional ou caixa-preta



Outras conotações do teste funcional

Teste funcional também é visto, na prática, como o teste por meio da interface, end-to-end, validação, etc

No fundo, é o teste do sistema como um todo por meio de sua interface, seja ela uma GUI, seja um conjunto de operações de uma API

Esta conotação surge do fato de que o teste funcional baseia-se na especificação (requisitos), e não no código

Outras conotações do teste funcional

Algumas pessoas não consideram o teste de unidade como teste funcional pois ele é feito pelos desenvolvedores, que têm acesso ao código e inconscientemente vão utilizar informações do código para gerar o teste

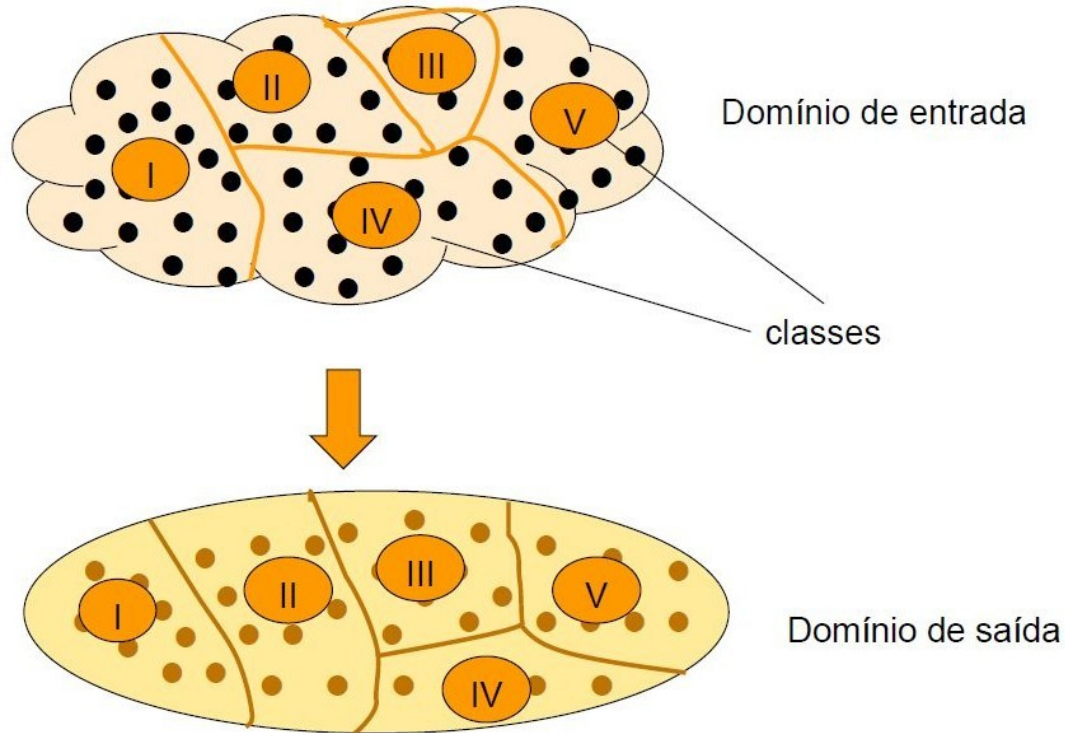
Entretanto, muitos desenvolvedores desenvolvem o teste antes mesmo de escrever o código, e para isso é necessário se basear em uma especificação

Basear-se na especificação e não no código é importante para que todas as regras sejam consideradas e omissões sejam detectadas

Critérios da técnica de teste funcional

- Classe de Equivalência
- Análise de valor-limite
- Tabela de decisão

Classes de Equivalência



Classes de Equivalência

- A partir das condições de entrada de dados identificadas na especificação, divide-se o domínio de entrada e de saída em classes de equivalência
- As classes de equivalência podem ser caracterizadas como válidas ou inválidas

Classes de Equivalência

- Em seguida, seleciona-se o menor número possível de casos de teste, baseando-se na hipótese que um elemento de uma dada classe seria representativo da classe toda, sendo que para cada uma das classes inválidas deve ser gerado um caso de teste distinto.

Classes de Equivalência

- O uso de particionamento permite examinar os requisitos mais sistematicamente e restringir o número de casos de teste existentes.
- Uma classe de equivalência representa um conjunto de estados válidos e inválidos para uma dada condição de entrada
- Classes de equivalência são disjuntas
- A união das classes de equivalência = domínio completo

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos	
PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	

Quais são as classes aqui?

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos	
PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	

Nas despesas com educação (ensino fundamental, médio, superior, o que engloba a graduação), o limite de dedução foi de R\$ 3.561,50 por dependente.

IRPF (uma simplificação) - CT1

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	9000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2
Total a pagar ou a receber:	

IRPF (uma simplificação) - CT2

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	12000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2
Total a pagar ou a receber:	

IRPF (uma simplificação) - CT3

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	15000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2
Total a pagar ou a receber:	

IRPF (uma simplificação) - CT3

Entrada	Valor
Rend. tributáveis recebidos	12000
PJ:	0
Rend. isentos e não-tributáveis:	0
Imposto pago (retido na fonte):	3000
Pagamentos efetuados (educação):	1500
Pagamentos efetuados (saúde):	900
Número de dependentes:	2
Total a pagar ou a receber:	

9000 vs 12000 vs 15000
(todos pertencem à mesma classe)
4500 / 6000 / 7500 por dependente (até o limite)

IRPF (uma simplificação) - CT3

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	4000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2
Total a pagar ou a receber:	

IRPF (uma simplificação) - CT93000 vs 12000 vs 15000

(todos pertencem à mesma classe)

4500 / 6000 / 7500 por dependente (acima do limite)

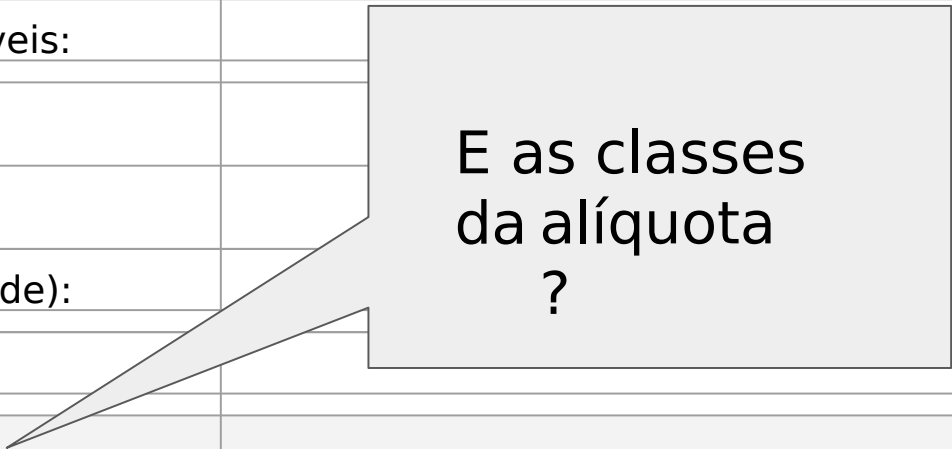
4000 ou 2000 por dependente (outra classe)

Entrada	Valor
Rend. tributáveis recebidos PJ:	120000
Rend. isentos e não-tributáveis:	20000
Imposto pago (retido na fonte):	30000
Pagamentos efetuados (educação):	4000
Pagamentos efetuados (saúde):	9000
Número de dependentes:	2

Total a pagar ou a receber:

Exemplo: IRPF (uma simplificação)

Entrada	Valor
Rend. tributáveis recebidos PJ:	
Rend. isentos e não-tributáveis:	
Imposto pago (retido na fonte):	
Pagamentos efetuados (educação):	
Pagamentos efetuados (saúde):	
Número de dependentes:	
Total a pagar ou a receber:	



E as classes da alíquota?

Exemplo: IRPF (uma simplificação)

	BASE DE CÁLCULO (R\$)	ALÍQUOTA (%)	PARCELA A DEDUZIR DO IRPF
Entrada			
Rend. PJ:	Até 1.903,98	isento	isento
Rend. Impos fonte) Pagar (educ: Pagar (saúde	De 1.903,99 até 2.826,65	7,5%	R\$142,80
	De 2.826,66 até 3.751,05	15%	R\$354,80
	De 3.751,06 até 4.664,68	22,5%	R\$636,13
	Acima de 4.664,68	27,5%	R\$869,36
Número			
Total a pagar ou a receber:			

Classes de Equivalência

Algumas diretrizes sobre como dividir o domínio de entrada em classes de equivalência:

- Quando há um intervalo de valores
- Uma classe válida para valores pertencentes ao intervalo
- Uma classe inválida para valores menores que o limite inferior
- Uma classe inválida para valores maiores que o limite superior

Quando há uma lista de valores válidos:

- Uma classe válida para cada valor incluído na lista
- Uma classe inválida para os valores foras da l

Classes de Equivalência

Algumas diretrizes sobre como dividir o domínio de entrada em classes de equivalência (cont...)

- Quando há restrições (expressão lógica; sintaxe; valor específico; compatibilidade com outras variáveis)
- Uma classe válida para os valores que satisfazem às restrições
- Uma classe inválida para os outros valores

Quando a saída é uma lista de valores distintos:

- Uma classe válida para cada entrada que provoca cada uma das saídas da lista
- Uma classe inválida para valores que não provocam cada uma das saídas da lista

Análise de Valor-Limite

- É um complemento ao critério Particionamento em Classes de Equivalência, sendo que os limites associados às condições de entrada são exercitados de forma mais rigorosa.
- Observações da prática profissional mostram que grande parte dos erros ocorre nas fronteiras do domínio de entrada

Análise de Valor-Limite

- Portanto, ao invés de selecionar qualquer elemento de uma classe, os casos de teste são escolhidos nas fronteiras das classes, pois nesses pontos se concentra um grande número de erros.
- O espaço de saída do programa também é particionado e são exigidos casos de teste que produzam resultados nos limites dessas classes de saída.

Análise de Valor-Limite

Diretrizes:

- Se os limites da condição de entrada forem a e b, projetar casos de teste para os valores de e imediatamente acima e abaixo de a e b.
- Se uma condição de entrada especifica vários valores, projetar casos de teste para os valores imediatamente acima e abaixo do valor mínimo e do valor máximo.
- Se as estruturas de dados internas do programa têm limites definidos na especificação do programa, deve-se criar casos de teste para exercitar a estrutura de dados no seu limite.

Exemplo 1 - Triângulo

- Escreva casos de teste para testar um algoritmo que recebe como entrada três números inteiros positivos que representam os tamanhos dos três lados de um triângulo, e retorna o tipo de triângulo formado (**equilátero, isósceles ou escaleno**).
- Se algum dos lados for negativo, o algoritmo deve informar que houve um erro porque um dos lados não tem valor válido (**lado inválido**)
- Se um dos lados tiver tamanho maior do que a soma dos outros dois lados, então o algoritmo deve informar que esses valores não são suficientes para formar um triângulo (**não forma triângulo**)

Exemplo 1 - Triângulo

Saídas
possíveis:

- Equilátero

Exemplo 1 - Triângulo

Saídas
possíveis:

- Equilátero
- Isósceles

Exemplo 1 - Triângulo

Saídas
possíveis:

- Equilátero
- Isósceles
- Escaleno

Exemplo 1 - Triângulo

Saídas possíveis:

- Equilátero
- Isósceles
- Escaleno
- Não forma triângulo

Exemplo 1 - Triângulo

Saídas possíveis:

- Equilátero
- Isósceles
- Escaleno
- Não forma triângulo
- Lado inválido

Exemplo 1 - Triângulo

Saídas

possíveis:

- Equilátero
- Isósceles
- Escaleno
- Não forma triângulo
- Lado inválido

Isósceles	Equilátero	Não forma triângulo
	Escaleno	Lado inválido

Exemplo 1 - Triângulo

Classes de equivalência:

- Equilátero (C1)
- IsóscelesAB (C2)
- IsóscelesAC (C3)
- IsóscelesBC (C4)
- Escaleno (C5)
- NFTAB (C6)
- NFTAC (C7)
- NFTBC (C8)
- LIA (C9)
- LIB (C10)
- LIC (C11)

Triângulo

Classes de equivalência:

- Equilátero (C1)
- IsóscelesAB (C2)
- IsóscelesAC (C3)
- IsóscelesBC (C4)
- Escaleno (C5)
- NFTAB (C6)
- NFTAC (C7)
- NFTBC (C8)
- LIA (C9)
- LIB (C10)
- LIC (C11)

[illegible]

Exemplo 1 - Triângulo

Classes de equivalência:

- Equilátero (C1)
- IsóscelesAB (C2)
- IsóscelesAC (C3)
- IsóscelesBC (C4)
- Escaleno (C5)
- NFTAB (C6)
- NFTAC (C7)
- NFTBC (C8)
- LIA (C9)
- LIB (C10)
- LIC (C11)

L A	L B	L C	Res. Esperado	Classe Equiv.
5	5	5	Equilátero	C1
5	5	4	Isósceles	C2
5	4	5	Isósceles	C3
4	5	5	Isósceles	C4
4	5	6	Escaleno	C5
2	2	1 0	Não forma	C6
2	1 0	2	Não forma	C7
1 0	2	2	Não forma	C8
-2	3	3	Lado Inválido	C9
3	-2	3	Lado Inválido	C10
3	3	-2	Lado Inválido	C11

Exemplo 1 -

Triângulo

Classes de Equivalência (alternativa: pensando em cada elemento da entrada)

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)

Exemplo 1 -

Triângulo

Classes de

Equivalência

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)

Exemplo 1 -

Triângulo

Classes de

Equivalência

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)
L3	$L3 > 0$ (C3)	$L3 \leq 0$ (C9)

Exemplo 1 -

Triângulo

Classes de

Equivalência

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)
L3	$L3 > 0$ (C3)	$L3 \leq 0$ (C9)
L1,L2,L3	Equilátero (C4): $L1 == L2 == L3$ Isósceles (C5): $L1 == L2 \neq L3$, $L2 == L3 \neq L1$, $L1 == L3 \neq L2$ Escaleno (C6): $L1 \neq L2 \neq L3 \neq L1$	

Exemplo 1 -

Triângulo

Classes de

Equivalência

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)
L3	$L3 > 0$ (C3)	$L3 \leq 0$ (C9)
L1,L2,L3	Equilátero (C4): $L1 == L2 == L3$ Isósceles (C5): $L1 == L2 \neq L3$, $L2 == L3 \neq L1$, $L1 == L3 \neq L2$ Escaleno (C6): $L1 \neq L2 \neq L3 \neq L1$	Não forma triângulo (C10): $L1 \geq L2 + L3$ $L2 \geq L1 + L3$ $L3 \geq L1 + L2$

Exemplo 1 -

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)
L3	$L3 > 0$ (C3)	$L3 \leq 0$ (C9)
L1,L2,L3	$L1 < L2 + L3$ (C11) $L2 < L1 + L3$ $L3 < L1 + L2$	Não forma triângulo (C10): $L1 \geq L2 + L3$ $L2 \geq L1 + L3$ $L3 \geq L1 + L2$
Equilátero	$L1 = L2$ e $L2 = L3$ (C4)	
Isósceles	$L1 = L2 \neq L3$, $L2 = L3 \neq L1$, $L1 = L3 \neq L2$ (C5)	
Escaleno	$L1 \neq L2 \neq L3 \neq L1$ (C6)	

Exemplo 1 -

Triângulo

Classes de

Equivalência

Entrada	Classes válidas	Classes inválidas
L1	$L1 > 0$ (C1)	$L1 \leq 0$ (C7)
L2	$L2 > 0$ (C2)	$L2 \leq 0$ (C8)
L3	$L3 > 0$ (C3)	$L3 \leq 0$ (C9)
L1,L2,L3	Equilátero (C4): $L1 == L2 == L3$ Isósceles (C5): $L1 == L2 \neq L3$, $L2 == L3 \neq L1$, $L1 == L3 \neq L2$ Escaleno (C6): $L1 \neq L2 \neq L3 \neq L1$	Não forma triângulo (C10): $L1 \geq L2 + L3$ $L2 \geq L1 + L3$ $L3 \geq L1 + L2$

[illegible]

Exemplo 1 - Triângulo

[illegible]

Triên cứu

[illegible]

— Δ —

[illegible]

Exemplo 1 - Triângulo

[illegible]

Exemplo 1 -

ID	Entrada	Saída esperada	Classe de equivalência
1	5,5,5	Equilátero	C1,C2,C3,C4
2	5,5,6	Isósceles	C1,C2,C3,C5.1
3	6,5,5	Isósceles	C1,C2,C3,C5.2
4	6,5,6	Isósceles	C1,C2,C3,C5.3
5	4,5,6	Escaleno	C1,C2,C3,C6
6	-2,4,5	ERRO - valor inválido	C7

Exemplo 1 - Triângulo

ID	Entrada	Saída esperada	Classe de equivalência
1	5,5,5	Equilátero	C1,C2,C3,C4
2	5,5,6	Isósceles	C1,C2,C3,C5.1
3	6,5,5	Isósceles	C1,C2,C3,C5.2
4	6,5,6	Isósceles	C1,C2,C3,C5.3
5	4,5,6	Escaleno	C1,C2,C3,C6
6	-2,4,5	ERRO – valor inválido	C7
7	4,-2,5	ERRO – valor inválido	C8

Exemplo 1 - Triângulo

ID	Entrada	Saída esperada	Classe de equivalência
1	5,5,5	Equilátero	C1,C2,C3,C4
2	5,5,6	Isósceles	C1,C2,C3,C5.1
3	6,5,5	Isósceles	C1,C2,C3,C5.2
4	6,5,6	Isósceles	C1,C2,C3,C5.3
5	4,5,6	Escaleno	C1,C2,C3,C6
6	-2,4,5	ERRO – valor inválido	C7
7	4,-2,5	ERRO – valor inválido	C8
8	4,5,-2	ERRO – valor inválido	C9
9	10, 2, 3	Não forma triângulo	C10.1
10	2, 10, 3	Não forma triângulo	C10.2
11	2, 3, 10	Não forma triângulo	C10.3

Exemplo 1 - Triângulo

Análise de Valor-
Limite

Entrada	Classes válidas	Classes inválidas
L1	L1=1 (v1)	L1=0 (v5), L1=-1 (v6)
L2	L2=1 (v2)	L2=0 (v7), L2=-1 (v8)
L3	L3=1 (v3)	L3=0 (v9), L3=-1 (v10)

Exemplo 1 - Triângulo

Análise de Valor-

Limite

Entrada	Classes válidas	Classes inválidas
L1	L1=1 (v1)	L1=0 (v5), L1=-1 (v6)
L2	L2=1 (v2)	L2=0 (v7), L2=-1 (v8)
L3	L3=1 (v3)	L3=0 (v9), L3=-1 (v10)
L1,L2,L3	Equilátero (1,1,1) (v4)	Não forma triângulo: L1 = L2 + L3 (v11) L2 = L1 + L3 (v12) L3 = L1 + L2 (v13)

Exemplo 1 -

ID	Entrada	Saída esperada	Valor-limite
12	0, 3, 4	ERRO - valor inválido	v5
13	-1, 3, 4	ERRO - valor inválido	v6
14	1, 2, 2	Isósceles	v1
15	3, 0, 4	ERRO - valor inválido	v7
16	3, -1, 4	ERRO - valor inválido	v8
17	2, 1, 2	Isósceles	v2
18	2, 3, 0	ERRO - valor inválido	v9
19	2, 3, -1	ERRO - valor inválido	v10
20	2, 2, 1	Isósceles	v3
21	1, 1, 1	Equilátero	v4
22	5, 2, 3	Não forma triângulo	v11

Exemplo 1 - Triângulo

[illegible]

Análise de causa-efeito

- Partição em classes de equivalência e Análise de Valores-Limite:
 - Não levam em conta combinações de valores difícil testar situações em que diferentes combinações levam a diferentes saídas do sistema

Análise de causa-efeito

- Útil quando especificações são representadas como estruturas de decisão:
 - Conjuntos de condições sobre valores de entradas e as ações correspondentes do sistema
 - Causas: condições de entrada (valor lógico)
 - Efeitos: ações realizadas em resposta às diferentes condições de entrada
- Modelos de teste:
 - tabelas de decisão

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles											
Equilátero				X							
Escaleno											
Impossível					X						

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles											
Equilátero				X							
Escaleno											
Impossível					X	X					

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles							X				
Equilátero				X							
Escaleno											
Impossível					X	X					

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles							X				
Equilátero				X							
Escaleno											
Impossível					X	X		X			

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles							X		X		
Equilátero				X							
Escaleno											
Impossível					X	X		X			

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles							X		X	X	
Equilátero				X							
Escaleno											
Impossível					X	X		X			

Condições											
$L1 < L2 + L3$	F	V	V	V	V	V	V	V	V	V	V
$L2 < L1 + L3$		F	V	V	V	V	V	V	V	V	V
$L3 < L2 + L3$			F	V	V	V	V	V	V	V	V
$L1 = L2$				V	V	V	V	F	F	F	F
$L2 = L3$				V	V	F	F	V	V	F	F
$L1 = L3$				V	F	V	F	V	F	V	F
Saídas											
Não é triângulo	X	X	X								
Isósceles							X		X	X	
Equilátero				X							
Escaleno											X
Impossível					X	X		X			

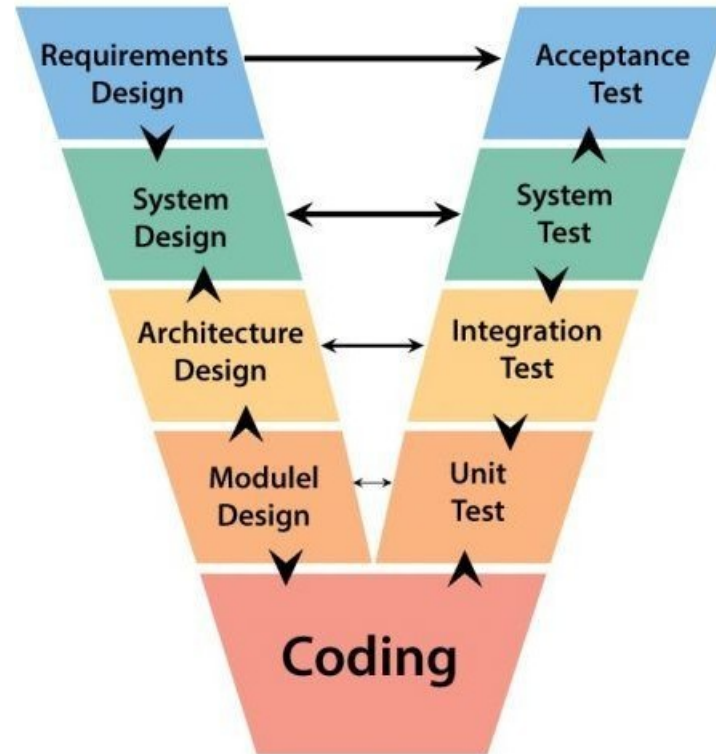
Benefícios da tabela de decisão

- Facilita a determinação de quais testes aplicar.
- Permite que se analise a especificação para determinar:
 - Redundâncias: duas regras iguais, i.e, mesmas causas levando aos mesmos efeitos
 - Contradições: duas regras com as mesmas causas levando a efeitos diferentes
 - Omissões: não há regras para todas as combinações de causas.
 - Redundâncias e contradições não são necessariamente erros: podem indicar concorrência.
 - Omissões podem indicar situações irrelevantes ou até mesmo impossíveis

Limitação das tabelas de decisão

- Explosão de combinações


Etapas típicas do teste de software



Teste funcional em cada etapa




Produto(s)	Excluir	Qtde	Valor Unitário	Total
 <div>Chico Bento (2021) - 02 Pião de Briga Pré-Venda!</div>		<div>+</div> <div>1</div>	R\$ 7,90	R\$ 7,90
<div>Custo de entrega:</div> <div>Informe seu CEP: <input type="text" value="19500-000"/><div>CALCULAR</div></div> <div><div><input checked="" type="radio"/> Impresso c/ Registro (13 a 23 dias úteis após manuseio) <input type="radio"/> PAC (11 a 21 dias úteis após manuseio) <input type="radio"/> SEDEX (5 a 7 dias úteis após manuseio)</div><div>R\$ 8,85 R\$ 15,12 R\$ 17,11</div></div>			Subtotal	R\$ 7,90
			Custo de Entrega	R\$ 8,85
			Subtotal de produto(s)	R\$ 16,75





<div>Possui um cupom de desconto?</div> <div><input type="text" value="Código do Cupom"/><div>UTILIZAR</div></div>	
Total geral	R\$ 16,75


[> AVANÇAR](#)

Teste funcional em cada etapa

Produto(s)	Excluir	Qtd	Valor Unitário	Total
 Chico Bento (2021) - 02 Pão de Brioche Pré-Venda!		<div>+ 1</div>	R\$ 7,90	R\$ 7,90
<div> Custo de entrega: Informe seu CEP: 19500-000 CALCULAR</div> <div><input checked="" type="radio"/> Impresso c/ Registro (13 a 23 dias úteis após manuseio) R\$ 8,85 <input type="radio"/> PAC (11 a 21 dias úteis após manuseio) R\$ 15,12 <input type="radio"/> SEDEX (5 a 7 dias úteis após manuseio) R\$ 17,11</div>			Subtotal	R\$ 7,90
			Custo de Entrega	R\$ 8,85
			Subtotal de produto(s)	R\$ 16,75



<div> Possui um cupom de desconto?</div> <div><input type="text" value="Código do Cupom"/></div> <div>UTILIZAR</div>	
Total geral	R\$ 16,75

 **AVANÇAR**

- calcularFrete(...)
- validarCupom(...)
- calcularDesconto(...)
- calcularValorCompra(...)
- verificarEstoque(...)
- ...