



Material de Apoio para Estudo

Análise de Valor Limite

Versão 1.0

Direitos Autorais

Copyright© Brazilian Software Testing Qualifications Board (doravante denominado BSTQB®)

BSTQB® é uma marca registrada da ABRAMTI Associação Brasileira de Melhoria em Ti.

BSTQB® é o Conselho Membro do ISTQB® International Software Testing Qualifications Board representando o Brasil nesta instituição.

Copyright©2023 autores da versão inicial (ordem alfabética): George Fialkovitz, Osmar Higashi e Stênio Viveiros.

Todos os direitos reservados. Os autores transferem os direitos autorais para o Brazilian Software Testing Qualifications Board (BSTQB®). Os autores (como detentores atuais de direitos autorais) e o BSTQB® (como futuro detentor dos direitos autorais) concordaram com as seguintes condições de uso:

- Este material foi produzido para apoiar o estudo do candidato interessado nos exames de certificação em Teste de Software do ISTQB®.
- Este material não pode ser comercializado.
- Extratos deste documento podem ser copiados se a fonte for reconhecida.
- Qualquer indivíduo ou grupo de indivíduos pode usar este material como base para artigos e livros, se os autores e o BSTQB® forem reconhecidos como a fonte e os proprietários dos direitos autorais

Histórico

Versão	Data	Observação
0.0	01/08/2023	Versão inicial
0.1	22/08/2023	Acrescentado um exemplo de partição de equivalência
1.0	30/08/2023	Lançamento

Sumário

Direitos Autorais	2
Histórico	3
Sumário	3
1 Introdução.....	4
1.1 Particionamento de Equivalência (EP).....	4
1.1.1 Como identificar uma Partição de Equivalência	5
1.1.2 Exemplo de Partição de Equivalência.....	5
2 Análise de Valor Limite (BVA)	7
2.1 Benefícios	7
2.2 Dificuldades e Limitações	8
2.3 Exemplos da utilização da Análise de Valor Limite	9
2.3.1 Análise de Valor Limite de dois valores.....	9
2.3.2 Análise de Valor Limite de três valores	10
2.4 Cobertura	11
3 Recapitulando.....	12
4 Referência	13

1 Introdução

A Análise de Valor Limite (*BVA - Boundary Value Analysis*) é uma técnica de teste de software com o principal objetivo de encontrar erros que ocorram quando os dados de entrada se aproximam dos valores limites de um campo, onde é mais provável que ocorram falhas no software.

Ela é baseada no conceito de partição de equivalência, que divide os dados de entrada em grupos representativos. Cada partição contém um conjunto de valores de entrada que são tratados de maneira equivalente pelo software.

1.1 Particionamento de Equivalência (EP)

Particionamento de Equivalência é uma técnica de teste de software que envolve a divisão do conjunto de dados de entrada em grupos ou partições, onde cada partição é tratada de maneira equivalente pelo software em teste. O objetivo é reduzir a quantidade de casos de teste necessários para cobrir todas as possíveis situações, aumentando a eficiência do processo de teste.

Cada grupo ou partição representa um conjunto de valores de entrada que são tratados de forma semelhante pelo software. O princípio subjacente é que, se um caso de teste funciona corretamente para um valor da partição, ele também funcionará corretamente para todos os outros valores nessa mesma partição.

O objetivo do Particionamento de Equivalência é reduzir a redundância nos casos de teste, garantindo que sejam testados cenários representativos, mas sem a necessidade de testar todas as combinações possíveis de entradas. Essa técnica é especialmente útil quando o conjunto de entrada é grande ou infinito.

Para criar Partições de Equivalência, devem-se identificar as seguintes categorias:

Partições Válidas: É o conjunto que contém dados de entrada que devem ser aceitos e tratados corretamente pelo software. Por exemplo, se um campo aceita apenas números inteiros entre 1 e 100, um conjunto válido pode ser os valores 10, 15 e 20.

Partições Inválidas: É o conjunto que contém dados de entrada que não devem ser aceitos ou que devem ser rejeitados pelo software. Continuando o exemplo anterior, uma partição inválida pode ser valores menores que 1 ou maiores que 100.

Valores Limite: É o conjunto que contém dados de entrada nos limites das restrições do sistema. No exemplo dado, os valores limítrofes seriam 1 e 100, pois são os extremos da faixa permitida.

O objetivo é garantir que pelo menos um teste seja executado para cada partição de equivalência, representando o comportamento esperado para aquela categoria. Dessa forma, a detecção de erros se torna mais eficiente e efetiva.

Ao aplicar a técnica de Particionamento de Equivalência em conjunto com outras técnicas de teste, como a Análise de Valor Limite, é possível alcançar uma cobertura mais completa e sistemática dos cenários de teste, otimizando o processo de teste de software e aumentando a probabilidade de encontrar bugs antes que o software seja implantado em produção.

1.1.1 Como identificar uma Partição de Equivalência

A identificação de partição de equivalência é um passo crucial no processo de aplicação da técnica de teste de software chamada Análise de Valor Limite. Para identificar as partições de equivalência relevantes, siga os passos abaixo:

Entender os requisitos e restrições: Leia e compreenda os requisitos do software, especialmente aqueles relacionados aos parâmetros de entrada que serão testados. Certifique-se de entender as faixas permitidas, limites, tipos de dados aceitos e outras restrições aplicáveis.

Agrupar o conjunto de entrada: Identifique as diferentes categorias ou grupos de entradas que possuem um comportamento similar. Agrupe as entradas que produzam resultados equivalentes ou que sejam tratadas da mesma maneira pelo software.

Identificar os valores limites: Identifique os limites das partições de equivalência. Esses são os valores nas extremidades das partições ou que estejam próximos dos limites. Por exemplo, no mesmo campo que aceita números de 1 a 100, os valores limites seriam 1 e 100.

Definir as partições válidas e inválidas: Com base nos requisitos e nos grupos identificados, determine quais conjuntos de dados devem ser considerados válidos e quais devem ser considerados inválidos. Por exemplo, se um campo só aceita números entre 1 e 100, uma partição válida conteria valores entre 1 e 100, enquanto a partição inválida conteria valores menores que 1 ou maiores que 100.

Lembre-se de que a identificação adequada das partições de equivalência é fundamental para o sucesso da BVA, pois isso determinará quais cenários serão testados e quais resultados são esperados. Além disso, é importante verificar se todas as combinações de partições foram cobertas pelos casos de teste para garantir uma cobertura abrangente do software em teste.

1.1.2 Exemplo de Partição de Equivalência

Vamos supor que você está trabalhando em um aplicativo de notas e quer testar a funcionalidade de adicionar uma nova nota.

Funcionalidade: Adicionar Nova Nota

Entradas:

Título: string alfanumérica de no máximo 20 caracteres

Conteúdo da nota: string alfanumérica de no máximo 100 caracteres.

Agora, para aplicar a partição de equivalência, podemos dividir as entradas em partições:

Partição	Título	Conteúdo
1	Válido (1 a 20 caracteres)	Válido (1 a 100 caracteres)
2	Inválido (mais de 20 caracteres)	Válido (1 a 100 caracteres)
3	Válido (1 a 20 caracteres)	Inválido (mais de 100 caracteres)
4	Inválido (mais de 20 caracteres)	Inválido (mais de 100 caracteres)

Agora, para testar a funcionalidade, você precisaria escolher pelo menos um valor de cada partição de equivalência para testar. Isso garantirá que você está cobrindo uma variedade de cenários relevantes.

Por exemplo, você poderia usar:

Partição 1:

	Caracteres	Resultado	Texto usado
Título	16	Válido (<20)	<i>Lista de Compras</i>
Conteúdo	20	Válido (<100)	<i>Comprar Leite e Ovos</i>

Partição 2:

	Caracteres	Resultado	Texto usado
Título	81	Inválido (>20)	<i>Esta é uma nota com um título muito longa que excede os 20 caracteres permitidos.</i>
Conteúdo	20	Válido (<100)	<i>Comprar Leite e Ovos</i>

Partição 3:

	Caracteres	Resultado	Texto usado
Título	16	Válido (<20)	<i>Lista de Compras</i>
Conteúdo	129	Inválido (>100)	<i>Esta é uma nota com um conteúdo muito longo que excede os cem caracteres permitidos e o resultado deve ser inválido para o campo.</i>

Partição 4:

	Caracteres	Resultado	Texto usado
Título	81	Inválido (>20)	<i>Esta é uma nota com um título muito longa que excede os 20 caracteres permitidos.</i>
Conteúdo	129	Inválido (>100)	<i>Esta é uma nota com um conteúdo muito longo que excede os cem caracteres permitidos e o resultado deve ser inválido para o campo.</i>

Dessa forma, você está testando vários casos relevantes e garantindo que sua funcionalidade de adicionar notas seja abrangente.

2 Análise de Valor Limite (BVA)

Para aplicar a Análise de Valor Limite, são selecionados valores de teste que estejam nos extremos dessas partições de equivalência. Em vez de testar todos os valores possíveis, a Análise de Valor Limite identifica os valores limites e testa-os, pois são nesses pontos que as falhas são mais propensas a ocorrer.

As principais etapas para realizar a BVA são:

Identificar as Partições de equivalência: ver 1.1.1

Priorizar valores limites e partições inválidas: Dê prioridade ao teste dos valores limites e das partições inválidas, pois são nesses pontos onde é mais provável encontrar erros no software. Os valores próximos aos limites têm maior probabilidade de causar problemas e devem ser testados com mais rigor.

Definir casos de teste para cada partição de equivalência: Com base nas partições de equivalência identificadas, crie casos de teste que representem cada categoria de entrada. Certifique-se de incluir os valores limites e valores próximos aos limites em seus casos de teste.

A BVA é uma técnica eficiente e eficaz, pois permite uma cobertura significativa de testes com um número relativamente pequeno de casos de teste. Além disso, ajuda a detectar erros em pontos críticos próximos aos limites, onde muitos bugs podem estar presentes. No entanto, é importante lembrar que a BVA deve ser usada em conjunto com outras técnicas de teste para obter uma cobertura completa e abrangente do software.

2.1 Benefícios

A Análise de Valor Limite é uma técnica valiosa de teste de software que traz diversos benefícios, ajudando a melhorar a qualidade do software e otimizando os esforços de teste. Alguns dos principais benefícios são:

Redução do número de casos de teste: A BVA foca nos valores extremos das partições de equivalência, permitindo testar apenas os valores limites relevantes em vez de todas as combinações possíveis de entrada. Isso leva a uma significativa redução na quantidade de casos de teste necessários para atingir uma cobertura abrangente.

Aumento da cobertura de testes: Ao testar os valores de entrada próximos aos limites das partições de equivalência, a BVA garante que cenários críticos sejam testados de forma eficiente. Isso aumenta a cobertura de testes e ajuda a identificar erros que poderiam passar despercebidos em testes convencionais.

Detecção precoce de erros: A BVA se concentra nos pontos onde é mais provável que ocorram falhas no software. Ao testar os valores limites, a técnica ajuda a identificar erros de programação, validação e comportamento do sistema mais rapidamente, permitindo correções antes que o software seja implantado em produção.

Identificação de problemas em valores extremos: Muitos erros e problemas de software tendem a ocorrer quando os dados de entrada se aproximam dos valores limites. A BVA

permite que esses pontos críticos sejam testados com mais atenção, aumentando a probabilidade de encontrar e corrigir problemas relevantes.

Melhoria da qualidade do produto final: Ao identificar e corrigir erros durante a fase de teste, a BVA contribui para a entrega de um produto final de maior qualidade. Isso resulta em menor chance de falhas em produção, reduzindo custos de correções pós-lançamento e melhorando a satisfação do usuário.

Otimização do processo de teste: A BVA oferece uma abordagem mais focada e direcionada aos testes, permitindo um uso mais eficiente dos recursos e do tempo disponíveis para a equipe de teste. Isso é especialmente relevante em projetos com prazos apertados ou recursos limitados.

A BVA é uma técnica poderosa para melhorar a eficácia e a eficiência do teste de software, tornando-o mais abrangente e identificando problemas críticos em valores extremos. Ao ser aplicada em conjunto com outras técnicas de teste, ela contribui para a entrega de um software de maior qualidade e confiabilidade.

2.2 Dificuldades e Limitações

A Análise de Valor Limite é uma técnica de teste de software eficaz para identificar erros em valores próximos aos limites das partições de equivalência. No entanto, ela também possui algumas limitações, que incluem:

Complexidade de cenários não lineares: A BVA pode se tornar menos eficaz em cenários em que as relações entre as entradas não são lineares. Se as interações entre as variáveis não puderem ser modeladas facilmente, a aplicação da BVA pode não ser tão direta.

Dificuldade em identificar partição de equivalência: Em sistemas complexos, pode ser difícil definir adequadamente as partições de equivalência e identificar seus limites de forma precisa. A criação de partições de equivalência pode se tornar um desafio quando existem muitos parâmetros de entrada envolvidos.

Dependência de outras técnicas de teste: A BVA é mais eficaz quando combinada com outras técnicas de teste, como testes funcionais e testes de integração. Por si só, ela não cobre todos os aspectos do teste de software.

Foco em limites específicos: A BVA concentra-se principalmente em testar os valores exatos dos limites, negligenciando os valores próximos a esses limites. Embora os testes de valor limite sejam importantes, não devem ser a única abordagem de teste.

Exigência de conhecimento do domínio: A aplicação eficaz da BVA requer um bom entendimento do domínio do problema e das regras de negócio envolvidas. Sem esse conhecimento, pode ser difícil identificar os valores limites relevantes.

Tamanho do conjunto de dados: A BVA pode gerar um grande número de casos de teste, especialmente em sistemas com muitos parâmetros de entrada. Isso pode aumentar o esforço de teste e a dificuldade na análise dos resultados.

Limitações em sistemas não determinísticos: Em sistemas que apresentam comportamento não determinístico ou onde os resultados são influenciados por fatores

externos, a BVA pode não ser tão efetiva, pois ela se baseia em cenários previsíveis e determinísticos.

Apesar dessas limitações, a BVA ainda é uma técnica valiosa para melhorar a cobertura de teste, identificar problemas em valores limites e otimizar os esforços de teste em várias situações. Ela deve ser utilizada em conjunto com outras abordagens de teste para garantir uma cobertura abrangente do sistema em desenvolvimento.

2.3 Exemplos da utilização da Análise de Valor Limite

2.3.1 Análise de Valor Limite de dois valores

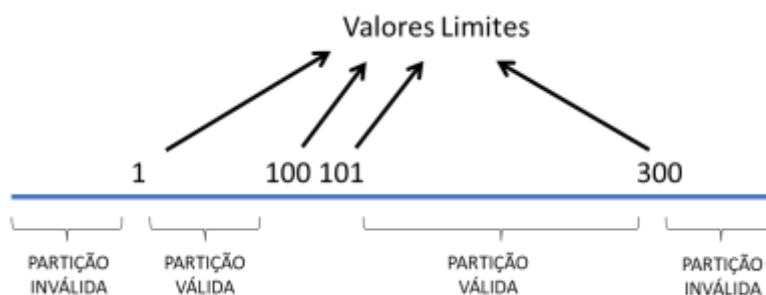
No BVA de 2 valores, para cada valor de limite há dois itens de cobertura: esse valor de limite e seu vizinho mais próximo pertencente à partição adjacente. Para atingir 100% de cobertura com o BVA de 2 valores, os casos de teste devem executar todos os itens de cobertura, ou seja, todos os valores de limite identificados. A cobertura é medida como o número de valores de limite que foram executados, dividido pelo número total de valores de limite identificados, e é expressa como uma porcentagem.

Vamos considerar um exemplo simples para ilustrar a Análise de Valor Limite para dois valores.

A velocidade de uma sonda submarina é determinada pela profundidade que ela está. A tabela abaixo mostra a relação entre a profundidade (metros) e a velocidade máxima (nós). A qualquer momento o operador poderá parar a sonda.

Profundidade (metros)	Velocidade (nós)
1 a 100	40
101 a 300	30

1) Dividir o conjunto de entrada em grupos (partição de equivalência).



2) Definir as entradas para cada partição.

- a) Valores Limite: 1, 100, 101 e 300
- b) Partições Válidas: 50 e 150
- c) Partições Inválidas: 0 e 301

3) Escrever os Casos de Teste.

Para se obter a cobertura necessária usando a BVA para 2 valores, precisamos testar 8 entradas.

Tipo	Caso de Teste	Valor de Entrada	Resultado esperado
Valor Limite	CT-1	1	A velocidade da sonda é de 40 nós
	CT-2	100	
	CT-3	101	A velocidade da sonda é de 30 nós
	CT-4	300	
Partição Inválida	CT-5	0	Erro de entrada apresentado
	CT-6	301	
Partição válida	CT-7	50	A velocidade da sonda é de 40 nós
	CT-8	150	A velocidade da sonda é de 30 nós

2.3.2 Análise de Valor Limite de três valores

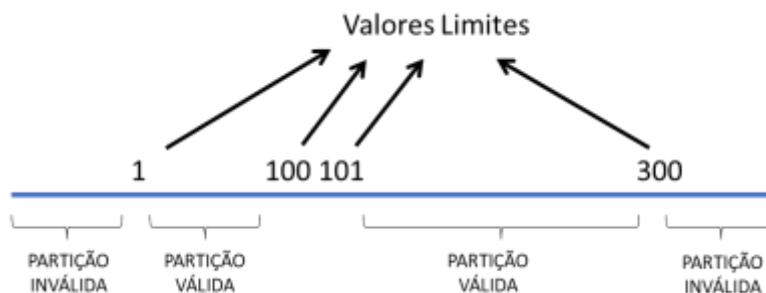
No BVA de 3 valores, para cada valor de limite há três itens de cobertura: esse valor de limite e seus dois vizinhos. Portanto, no BVA de 3 valores, alguns dos itens de cobertura podem não ser valores de limite. Para atingir 100% de cobertura com o BVA de 3 valores, os casos de teste devem executar todos os itens de cobertura, ou seja, os valores de limite identificados e seus vizinhos.

Vamos considerar o mesmo exemplo anterior.

A velocidade de uma sonda submarina é determinada pela profundidade que ela está. A tabela abaixo mostra a relação entre a profundidade (metros) e a velocidade máxima (nós). A qualquer momento o operador poderá parar a sonda.

Profundidade (metros)	Velocidade (nós)
1 a 100	40
101 a 300	30

1) Dividir o conjunto de entrada em grupos (partição de equivalência).



2) Definir as entradas para cada partição.

- a) Valores Limite: 1, 100, 101 e 300
- b) Partições Válidas: 2, 99, 102 e 299
- c) Partições Inválidas: 0 e 301

3) Escrever os Casos de Teste.

Para se obter a cobertura necessária usando a BVA para 3 valores, precisamos testar 10 entradas.

Tipo	Caso de Teste	Valor de Entrada	Resultado esperado
Valor Limite	CT-1	1	A velocidade da sonda é de 40 nós
	CT-2	100	
	CT-3	101	A velocidade da sonda é de 30 nós
	CT-4	300	
Partição Inválida	CT-5	0	Erro de entrada apresentado
	CT-6	301	
Partição válida	CT-7	1	A velocidade da sonda é de 40 nós
	CT-8	99	
	CT-9	102	A velocidade da sonda é de 30 nós
	CT-10	299	

2.4 Cobertura

A cobertura é medida é o percentual da quantidade de valores executados pelo total de valores identificados.

$$\frac{\text{Total de valores executados}}{\text{Total de valores identificados}}$$

3 Recapitulando...

A Análise de Valor Limite é uma técnica de teste de software essencial para otimizar a qualidade do produto final e maximizar a eficiência dos testes. Ao focar nos valores extremos e próximos aos limites das partições de equivalência, a BVA permite uma cobertura mais abrangente dos cenários críticos, onde erros são mais propensos a ocorrer.

Em resumo, a Análise de Valor Limite é uma ferramenta poderosa que otimiza a cobertura de testes, economiza recursos e aprimora a qualidade do software. Sua aplicação conjunta com outras técnicas de teste é fundamental para um processo de teste abrangente e eficiente, garantindo um produto final robusto e livre de falhas.

4 Referência

Ammann, P. and Offutt, J. (2016)

Introduction to Software Testing (2e), Cambridge University Press

Craig, R. and Jaskiel, S. (2002)

Systematic Software Testing, Artech House: Norwood MA

Koomen, T., van der Aalst, L., Broekman, B. and Vroon, M. (2006)

TMap Next for result-driven testing, UTN Publishers, The Netherlands

Myers, G. (2011)

The Art of Software Testing, (3e), John Wiley & Sons: New York NY

O'Regan, G. (2019)

Concise Guide to Software Testing, Springer Nature Switzerland

Pressman, R.S. (2019)

Software Engineering. A Practitioner's Approach, 9th ed., McGraw Hill