**Classe de Equivalência**

**Particionamento de equivalência**

Reduz um conjunto de entradas de grande (infinito) a um conjunto finito: pequeno, mas eficiente. Divide o domínio de entrada de um software (ou programa) em classes de dados a partir das quais os casos de teste podem ser derivados.

Classe de equivalência representa um conjunto de estados válidos ou inválidos para condições de entrada

* Duas ou mais classes são equivalentes, se elas produzem o mesmo resultado.
* Se uma situação funciona como o esperado, então podemos assumir que a outra também funciona.
* Devem conter entradas válidas e invalidas

Uma condição de entrada pode ser: um valor numérico; um intervalo de valores; um valor alfabético; etc.

**Exemplo:**

 Considere o Seguinte procedimento: Valida\_Nova\_Senha que  recebe como entrada uma senha e valida conforme as  regras abaixo. Defina as classes de equivalência e os casos de teste.

* A senha deve ter de 6 a 10 caracteres
* O primeiro carácter deve ser alfabético, numérico ou “?”
* Os outros caracteres podem ser quaisquer, desde que não sejam  caracteres de controle.
* A senha não pode existir num dicionário.

**Análise de valor limite**

A Análise de Valor Limite (Boundary Value Analysis – BVA) é uma técnica de projeto de casos de teste que complementa o particionamento de equivalência; Em vez de selecionar qualquer elemento de uma classe de equivalência, a BVA leva à seleção de casos de teste nas “extremidades” da classe:

* Use um ou dois casos de testes para cada classe de equivalência;
* O melhor deles são os valores limites;
* Incorretas não-igualdades causam falhas apenas nos limites;
* Programas que falham com valores não-limites normalmente falham com valores limites também;

Se um intervalo é delimitado por a e b, os casos de testes devem ser projetados com valores a e b logo acima e logo abaixo respectivamente

* Se a faixa de valor de entrada é 1 a 99
* Se a faixa de valor de entrada e R$ 1 a R$ 99
* Se o programa espera Maiúscula de A a Z
* Se o programa desenha linhas  com um tamanho definido
* Se os valores de entrada devem somar 180
* Se o programa necessita especificar o numero de entradas.

**Exemplos:**

 1) A senha deve ter entre 6 e 10 caracteres.

 If (senha > 6 and senha < 10) → erro >= 6 e <= 10

          loga

else

          exibe mensagem: “senha incorreta”

 2) Criar uma lista de 10 elementos e preencher com a

lista[10] → erro cria 10 elementos e preenche 9

for (i = 1; i< 10; i++)

          lista (i) = a

end for

 Considere uma Função que aceita como entradas de 4 a 6 valores inteiros de 2 dígitos cada maiores que 10. Aplique a técnica de analise do Valor Limite e defina os casos de teste para a função.