



Nome do Aluno: Antonio Josivaldo Dantas Filho

Unidade: Unidade 6 - Verificação & Validação

ES2 - AA-6 - Aplicação de Critérios de Teste

Para apresentar o Particionamento de Equivalência (PE) terá como base a especificação: “recebe como entrada três valores inteiros que são interpretados como o comprimento dos lados de um triângulo e exibe mensagem dizendo se o triângulo definido pelos três valores informados é isósceles, escaleno ou equilátero”, será apresentado a perspectivas do ponto de vista das saídas e entre parênteses das entradas.

Classes de equivalência:

- Triângulo equilátero ($\text{lado1} = \text{lado2} = \text{lado3}$)
- Triângulo Isósceles ($\text{lado1} = \text{lado2}$)
- Triângulo Isósceles ($\text{lado2} = \text{lado3}$)
- Triângulo Isósceles ($\text{lado1} = \text{lado3}$)
- Triângulo escalenos ($\text{lado1} <> \text{lado2} <> \text{lado3}$)
- Os três lados não formam um triângulo ($\text{lado1} \geq (\text{lado2} + \text{lado3})$)
- Os três lados não formam um triângulo ($\text{lado2} \geq (\text{lado1} + \text{lado3})$)
- Os três lados não formam um triângulo ($\text{lado3} \geq (\text{lado1} + \text{lado2})$)

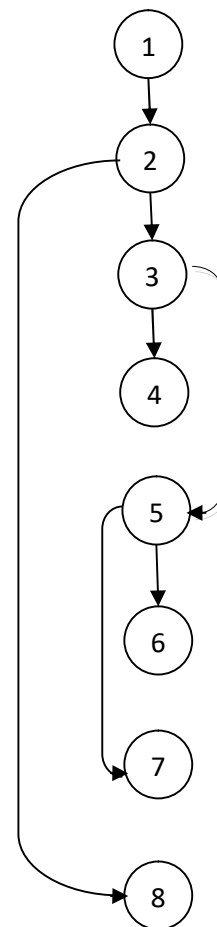
É apresentado então o conjunto de caso de teste na forma de tabela de modo finito e custo de tempo empregado razoável, e de acordo com os critérios estabelecidos anteriormente:

Entrada				Saída Esperada (Classes)
	Lado 1	Lado 2	Lado 3	
1	1	1	1	Triângulo equilátero
2	22	22	30	Triângulo Isósceles
3	6	5	5	Triângulo Isósceles
4	12	7	12	Triângulo Isósceles
5	5	7	9	Triângulo escalenos
6	50	10	20	Os três lados não formam um triângulo
7	1	3	2	Os três lados não formam um triângulo
8	2	20	21	Os três lados não formam um triângulo



Será apresentado o grafo de fluxo de controle representando o código do enunciado:

```
program triangulo;  
uses crt;  
var  
    lado1, lado2, lado3: integer;  
begin  
    clrscr;  
    1 { writeln ('digite o primeiro lado : ');  
      readln (lado1);  
      writeln ('digite o segundo lado : ');  
      readln (lado2);  
      writeln ('digite o terceiro lado : ');  
      readln (lado3);  
      //verificar se é possível formar triangulo;  
      2 { if ( lado1 < (lado2+lado3) ) and  
          ( lado2 < (lado1+lado3) ) and  
          ( lado3 < (lado1+lado2) ) then  
          begin //se for possível então:  
              //verificar se é equilátero  
              3 { if ( (lado1 = lado2) and  
                    (lado2 = lado3) ) then  
                  4 { writeln('triângulo equilátero')  
                    //verificar se é isósceles  
                  5 { else if ( ( lado1 = lado2) or  
                                ( lado1 = lado3) or  
                                ( lado2 = lado3) ) then  
                      6 { writeln('triângulo isósceles')  
                      else  
                      7 { // verificar se é triângulo escaleno  
                        writeln('triângulo escaleno');  
                    end  
                }  
            }  
        }  
    }  
    8 { else //não foi possível formar triângulo  
        writeln('os três lados não formam um triângulo');  
        readkey  
    }  
end.
```





A seguir é apresentado o teste de ramos que por definição estabelece como dados de teste exercitar todas as saídas verdadeiro e falso de todas as decisões, observando o grafo, é necessário gerar teste que causem as duas saídas verdadeiro e falso nos nós 2,3 e 5.

Entrada				Ramo		Saída Esperada (Classes)
	Lado 1	Lado 2	Lado 3	Nó	Booleano	
1	3	2	1	2	Falso	Os três lados não formam um triângulo
2	5	5	5	2	Verdadeiro	Triângulo equilátero
				3	Verdadeiro	
3	6	6	4	3	Falso	Triângulo Isósceles
				5	Verdadeiro	
4	7	8	9	5	Falso	Triângulo escalenos

Concluindo, foi de grande importância a realização do trabalho pois permitiu a aprendizagem de forma prática na verificação e validação. Os dois modos solicitados, Particionamento de equivalência e teste de ramos abrangeu bem de forma a atingir desde a documentação até o código compilado. Foi possível observar que o código está com bom funcionamento, mas é impossível garantir a margem zero de erros, pois necessitaria de um teste exaustivo onde não seria possível dado o intervalo possível que o usuário pode utilizar.