

## ATIVIDADE 1: IMPLEMENTAÇÃO DE CASOS DE SEMELHANÇA DE TRIÂNGULOS

Link GitHub: <https://github.com/filipetocchio/-matematica-computacional>

### Explicação da Lógica do Algoritmo

#### 1. Entrada de Dados dos Triângulos:

O algoritmo inicia solicitando ao usuário as medidas de dois triângulos, que incluem os lados e ângulos. As entradas de dados são ajustadas conforme o critério selecionado para verificar a semelhança (LAL, AA ou LLL), adequando-se às características necessárias para cada critério.

- Para o critério LAL, o algoritmo pede dois lados e o ângulo entre eles.
- Para o critério AA, são solicitados dois ângulos.
- Para o critério LLL, todos os três lados são requeridos.

#### 2. Critérios de Semelhança:

- **Critério LAL (Lado-Ângulo-Lado):** A semelhança é confirmada quando:
  - Os ângulos entre os lados correspondentes dos dois triângulos são congruentes.
  - Os lados ao redor desse ângulo mantêm uma relação de proporção constante entre si.
  - Exemplo: para triângulos com lados (lado1\_1, lado2\_1) e ângulo angulo\_1 em comparação a (lado1\_2, lado2\_2) e angulo\_2, verificamos a igualdade dos ângulos e a proporcionalidade entre  $\text{lado1\_1}/\text{lado1\_2} == \text{lado2\_1}/\text{lado2\_2}$ .
- **Critério AA (Ângulo-Ângulo):** A semelhança ocorre quando:
  - Dois ângulos dos triângulos são congruentes, garantindo que o terceiro ângulo também será congruente, pois a soma dos ângulos internos de um triângulo é sempre 180 graus.
  - A congruência dos dois ângulos confirma que os triângulos são semelhantes por este critério, sem necessidade de verificar os lados.
- **Critério LLL (Lado-Lado-Lado):** A semelhança é verificada quando:
  - Todos os lados dos dois triângulos possuem uma relação de proporção constante entre si.
  - Para os lados (lado1\_1, lado2\_1, lado3\_1) de um triângulo e (lado1\_2, lado2\_2, lado3\_2) do outro, os três devem satisfazer  $\text{lado1\_1}/\text{lado1\_2} == \text{lado2\_1}/\text{lado2\_2} == \text{lado3\_1}/\text{lado3\_2}$ .

#### 3. Verificação de Semelhança:

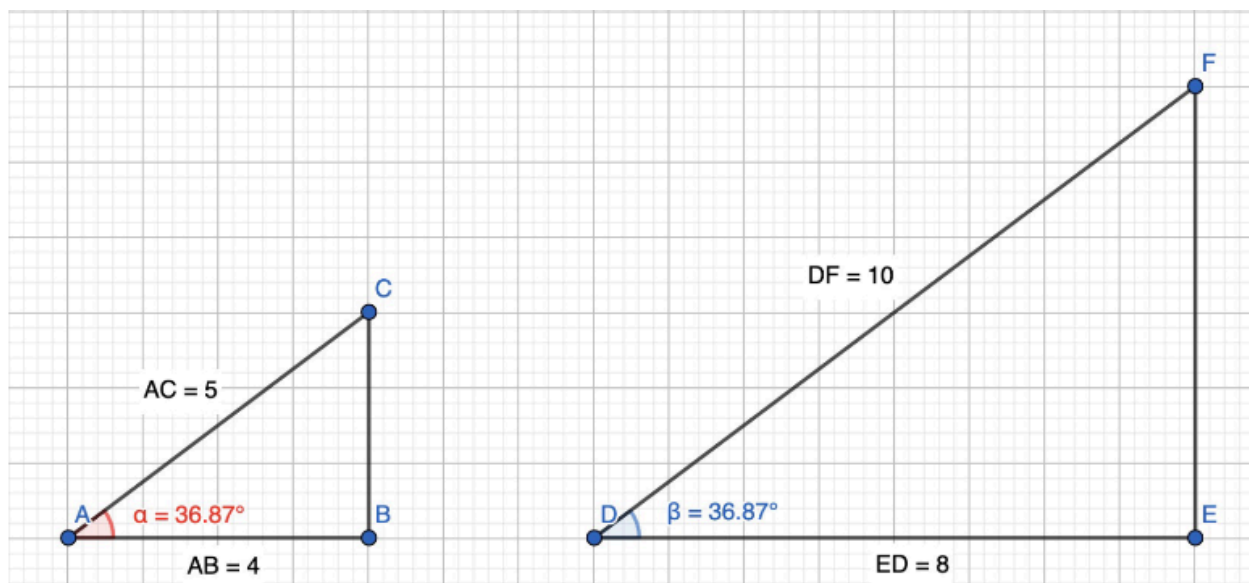
Após coletar os dados dos triângulos, a função principal de verificação identifica o critério e chama a função correspondente (`verificar_lal`, `verificar_aa` ou `verificar_III`). Cada função retorna True ou False, dependendo se a condição de semelhança é satisfeita. Se qualquer uma das verificações retornar True, o algoritmo considera os triângulos como semelhantes pelo critério em questão.

#### 4. Resultado Final:

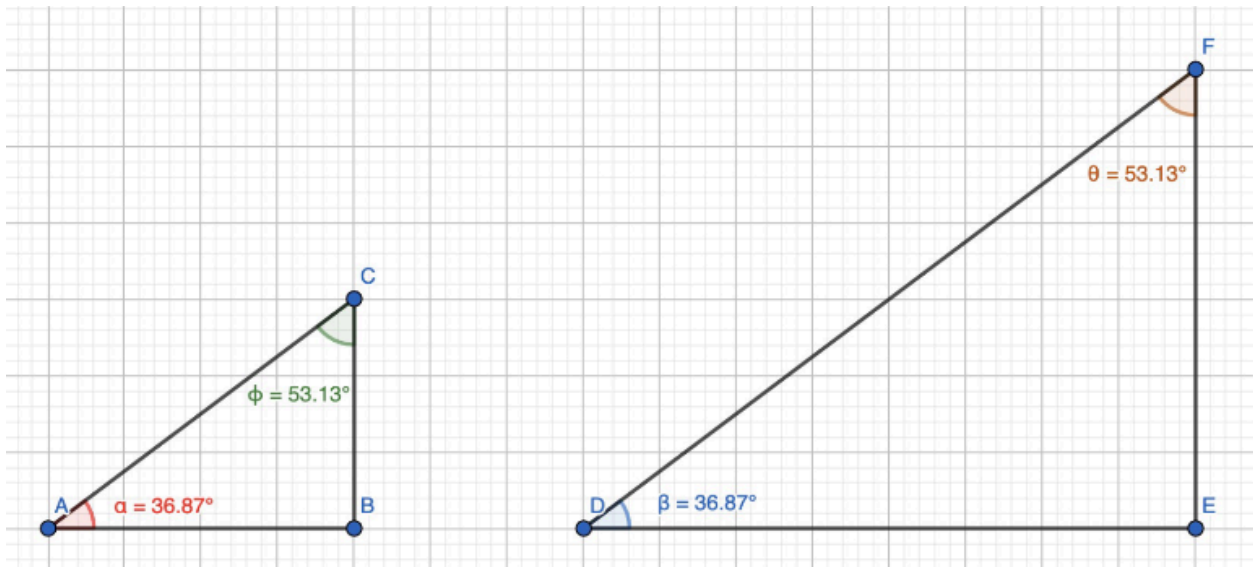
Se um critério de semelhança é validado, uma mensagem indicando o critério específico é exibida, afirmando que os triângulos são semelhantes. Caso contrário, o algoritmo informa que os triângulos não são semelhantes.

### ESBOÇOS DE CADA CASO NO GEOGEBRA :

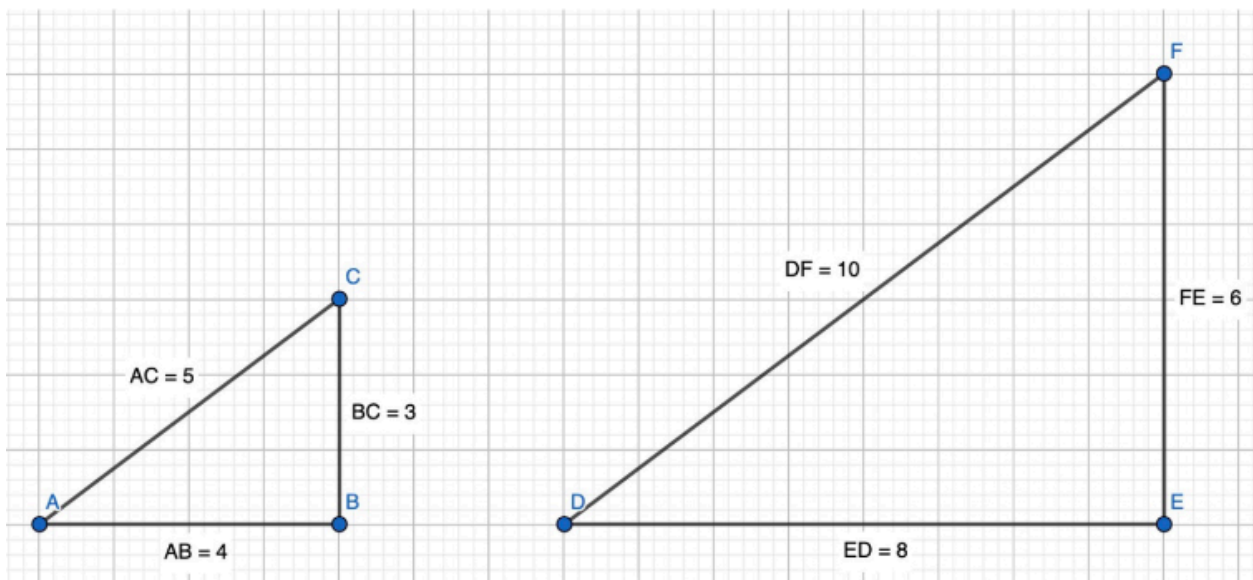
#### LAL (Lado-Ângulo-Lado):



## AA (Ângulo-Ângulo):



## LLL (Lado-Lado-Lado):



## Código em Python:

O código, além de estar aqui no docs, ele também se encontra no GitHub, fornecido neste mesmo arquivo e também fornecido do arquivo enviado da atividade.

```

1 # Funções de verificação de semelhança
2
3 def verificar_lal(lado1_1, lado2_1, angulo_1, lado1_2, lado2_2, angulo_2):
4     """Verifica se dois triângulos são semelhantes pelo critério LAL (Lado-Ângulo-Lado)."""
5     return angulo_1 == angulo_2 and lado1_1 / lado1_2 == lado2_1 / lado2_2
6
7 def verificar_aa(angulo1_1, angulo2_1, angulo1_2, angulo2_2):
8     """Verifica se dois triângulos são semelhantes pelo critério AA (Ângulo-Ângulo)."""
9     return (angulo1_1 == angulo1_2 and angulo2_1 == angulo2_2) or (angulo1_1 == angulo2_2 and angulo2_1 == angulo1_2)
10
11 def verificar_lll(lado1_1, lado2_1, lado3_1, lado1_2, lado2_2, lado3_2):
12     """Verifica se dois triângulos são semelhantes pelo critério LLL (Lado-Lado-Lado)."""
13     return lado1_1 / lado1_2 == lado2_1 / lado2_2 == lado3_1 / lado3_2
14
15 # Função para obter dados do triângulo
16 def obter_triangulo(criterio):
17     if criterio == "LAL":
18         lado1 = float(input("Digite o primeiro lado: "))
19         angulo = float(input("Digite o ângulo entre os lados: "))
20         lado2 = float(input("Digite o segundo lado: "))
21         return (lado1, angulo, lado2)
22
23     elif criterio == "AA":
24         angulo1 = float(input("Digite o primeiro ângulo: "))
25         angulo2 = float(input("Digite o segundo ângulo: "))
26         return (angulo1, angulo2)
27
28     elif criterio == "LLL":
29         lado1 = float(input("Digite o primeiro lado: "))
30         lado2 = float(input("Digite o segundo lado: "))
31         lado3 = float(input("Digite o terceiro lado: "))
32         return (lado1, lado2, lado3)
33
34 # Função principal para verificar semelhança
35 def verificar_semelhanca(triangulo1, triangulo2, criterio):
36     if criterio == "LAL":
37         return verificar_lal(*triangulo1, *triangulo2)
38     elif criterio == "AA":
39         return verificar_aa(*triangulo1, *triangulo2)
40     elif criterio == "LLL":
41         return verificar_lll(*triangulo1, *triangulo2)
42     else:
43         print("Critério inválido.")
44         return False
45
46 # Função para executar testes
47 def executar_testes():
48     print("Iniciando testes...\n")
49
50     # Teste LAL
51     print("Teste LAL: lados (3, 4), ângulo 60 vs lados (6, 8), ângulo 60")
52     assert verificar_lal(3, 4, 60, 6, 8, 60), "Erro no teste LAL!"
53
54     # Teste AA
55     print("Teste AA: ângulos (30, 60) vs ângulos (30, 60)")
56     assert verificar_aa(30, 60, 30, 60), "Erro no teste AA!"
57
58     # Teste LLL
59     print("Teste LLL: lados (3, 4, 5) vs lados (6, 8, 10)")
60     assert verificar_lll(3, 4, 5, 6, 8, 10), "Erro no teste LLL!"
61
62     print("Todos os testes passaram com sucesso!")
63
64 # Programa principal
65 criterio = input("Escolha o critério de semelhança (LAL, AA, LLL): ").upper()
66
67 print("Digite os valores para o primeiro triângulo:")
68 triangulo1 = obter_triangulo(criterio)
69
70 print("Digite os valores para o segundo triângulo:")
71 triangulo2 = obter_triangulo(criterio)
72
73 resultado = verificar_semelhanca(triangulo1, triangulo2, criterio)
74
75 if resultado:
76     print(f"Os triângulos são semelhantes pelo critério {criterio}.")
77 else:
78     print(f"Os triângulos não são semelhantes pelo critério {criterio}.")

```