
Lab 3 – Condicionais Encadeadas(if/elif)

Questão 1

Escreva um programa que leia as medidas dos três lados A, B, C de um triângulo qualquer. Em seguida, ele deve verificar se as medidas são válidas (condição de existência de um triângulo).

Como resultado, o programa deverá imprimir a seguinte mensagem:

Entradas: A, B, C

Area: X

Substitua as letras **A, B, C** pelos valores de entrada informados pelo usuário. Se as entradas forem válidas, substitua a letra **X** pelo valor da área, com até **três casas** decimais de precisão. Caso contrário, substitua a letra **X** pela seguinte mensagem: **invalida**

Dicas

1. O programa está incompleto. Substitua as expressões **___COMPLETE AQUI___** por comandos que façam o programa executar corretamente.
2. A solução proposta adota os seguintes passos:
 1. Ler o valor de cada lado do triângulo: A, B, C.
 2. Verificar se todos os valores fornecidos são **positivos**.
 1. Se verdade, verificar se as medidas fornecidas correspondem às de um triângulo, ou seja, se a soma dos dois lados menores é maior que o lado maior.
 2. O teste deve ser verdadeiro para **todas** as três combinações possíveis.
 3. Se todos os testes forem bem sucedidos, calcule a área e exiba a mensagem.
 4. Se pelo menos um teste falhar, apresente a mensagem de erro.

Exemplo de Entrada

1.0

2.0

3.0

Exemplo de Saída

Entradas: 1.0 , 2.0 , 3.0

Area: invalida

```
1 from math import *
2 a = float(input ("Lado 1: "))
3 b = float(input ("Lado 2: "))
4 c = float(input ("Lado 3: "))
5
6 print("Entradas:", a, ",", b, ",", c)
7
8 if (a > 0) and (b > 0) and (c > 0):
9     if ((a < b+c) and (b < a+c) and (c < a+b)):
10         s = (a + b + c) / 2.0
11         area = sqrt(s * (s-a) * (s-b) * (s-c))
12         print("Area:", round(area, 3))
13     else:
14         print("Area: invalida")
15 else:
16     print("Area: invalida")
```

Questão 2

Escreva um programa que leia **três números reais**, correspondentes às medidas dos lados de um triângulo, em ordem qualquer.

Como resultado, o programa deverá imprimir:

Entradas: A, B, C

Tipo de triangulo: X

Substitua as letras **A, B, C** pelos valores de entrada informados pelo usuário. Substitua a letra **X** por um dos seguintes valores:

- **equilatero**, se todos os três lados forem iguais;
- **isosceles**, se apenas dois lados forem iguais;
- **escaleno**, se nenhum par de lados forem iguais;
- **invalido**, se pelo menos um dos lados for negativo ou se os três lados não formarem um triângulo.

Dicas

1. Atenção para o uso de caracteres maiúsculos e minúsculos. **Não use acentos**, para evitar erros.
2. **Condição de existência** de um triângulo: a medida de qualquer um dos lados deve menor que a soma das medidas dos outros dois.

Exemplo de Entrada

1.0

2.0

3.0

Exemplo de Saída

Entradas: 1.0 , 2.0 , 3.0

Tipo de triangulo: invalido

```
1 a = float(input("1. Digite o valor do Lado 1: "))
2 b = float(input("2. Digite o valor do Lado 2: "))
3 c = float(input("3. Digite o valor do Lado 3: "))
4
5 print("Entradas: ", a, ",", b, ",", c)
6
7 if (a > 0) and (b > 0) and (c > 0):
8     if (a < b+c) and (b < a+c) and (c < a+b):
9         if (a == b) and (b == c):
10             print("Tipo de triangulo: equilatero")
11         elif (a != b) and (b != c):
12             print("Tipo de triangulo: escaleno")
13         else:
14             print("Tipo de triangulo: isosceles")
15     else:
16         print("Tipo de triangulo: invalido")
17 else:
18     print("Tipo de triangulo: invalido")
```

Questão 3

As cédulas do real começaram a entrar em circulação no Brasil em 1994, estampando animais da fauna brasileira, conforme mostra a tabela a seguir.

Escreva um programa que leia o **valor de uma cédula**. Como saída, determine o **nome do animal** representado no verso dela.

Validação dos dados:

Se o usuário inserir valores de entrada **inválidos**, imprima:

Entrada: X

Animal: Invalido

Se as entradas forem **válidas**, imprima:

Entrada: X

Animal: Y

Nas mensagens, substitua as letras **X** e **Y** pelos valores correspondentes.

Valor da cédula	Animal
R\$ 2	Tartaruga
R\$ 5	Garça
R\$ 10	Arara
R\$ 20	Mico-leão-dourado
R\$ 50	Onça-pintada
R\$ 100	Garoupa

Dicas

1. Utilize uma condição encadeada (**elif**) para testar cada tipo de cédula.
2. Leia o valor das cédulas como um número **inteiro**.
3. Use letras maiúsculas e minúsculas exatamente como mostra o enunciado. **Não use acentos**.

Exemplo de Entrada

20

Exemplo de Saída

Entrada: 20

Animal: Mico-leao-dourado

```
1 valor = int(input("1.Digite o valor da cedula: "))
2
3 if (valor == 2):
4     txt = "Tartaruga"
5 elif (valor == 5):
6     txt = "Garca"
7 elif (valor == 10):
8     txt = "Arara"
9 elif (valor == 20):
10    txt = "Mico-leao-dourado"
11 elif (valor == 50):
12    txt = "Onca-pintada"
13 elif (valor == 100):
14    txt = "Garoupa"
15 else:
16    txt = "Invalido"
17
18 print("Entrada:", valor)
19 print("Animal:", txt)
```

Questão 4

Elabore um programa que leia a **descrição do brasão** de uma das casas da série de TV "Guerra dos Tronos", conforme tabela abaixo. Como saída, determine o **nome da casa** correspondente.

- Se a entrada for uma descrição válida, então imprima:

Entrada: X

Casa: Y

- Se a entrada for uma casa não listada na tabela, imprima:

Entrada: X

Brasao invalido

Substitua a letra **X** pela entrada inserida e a letra **Y** pela casa correspondente.

Brasão	Casa
lobo	Stark
leao	Lannister
veado	Baratheon
dragao	Targaryen
rosa	Tyrell
sol	Martell
lula	Greyjoy
esfolado	Bolton
turta	Tully

Dicas

- Utilize condições encadeadas (**elif**).
- Use letras maiúsculas e minúsculas da mesma maneira como descrita no enunciado. **Não use acentos**.
- Lembre-se que **"x"** (com aspas) indica o caractere xis, ao passo que **x** (sem aspas) indica o nome de uma variável chamada **x**.

Exemplo de Entrada

rosa

Exemplo de Saída

Entrada: rosa

Casa: Tyrell

```
1 brasao = input("1.Digite a respectiva descricao de seu brasao: ")
2
3 print("Entrada: " + brasao)
4
5 if (brasao == "lobo"):
6     print("Casa: Stark")
7 elif (brasao == "leao"):
8     print("Casa: Lannister")
9 elif (brasao == "veado"):
10    print("Casa: Baratheon")
11 elif (brasao == "dragao"):
12    print("Casa: Targaryen")
13 elif (brasao == "rosa"):
14    print("Casa: Tyrell")
15 elif (brasao == "sol"):
16    print("Casa: Martell")
17 elif (brasao == "lula"):
18    print("Casa: Greyjoy")
19 elif (brasao == "esfolado"):
20    print("Casa: Bolton")
21 elif (brasao == "torta"):
22    print("Casa: Tully")
23 else:
24    print("Brasao invalido")
```


Questão 5

Escreva um programa que determine o **valor total** a ser pago pela conta de energia elétrica, com base nas seguintes entradas:

1. O consumo de energia (em kWh); e
2. O tipo de instalação (**R** para residências, **I** para indústrias, e **C** para comércios).

Use a tabela a seguir para calcular o valor devido:

Preço por tipo e faixa de consumo		
Tipo	Consumo (em kWh)	Preço (por kWh)
Residencial	Até 500	R\$ 0,44
	Acima de 500	R\$ 0,65
Comercial	Até 1000	R\$ 0,55
	Acima de 1000	R\$ 0,60
Industrial	Até 5000	R\$ 0,55
	Acima de 5000	R\$ 0,60

Validação dos dados:

Se o usuário inserir valores de entrada **inválidos**, imprima:

Entradas: X kWh e tipo Y

Dados invalidos

Se as entradas forem válidas, imprima:

Entradas: X kWh e tipo Y

Valor total: R\$ Z

Nas mensagens de saída, substitua as letras **X**, **Y** e **Z** pelos valores correspondentes.

Dicas

1. Use a estrutura de programação **if-elif-else**.
2. Atenção para as letras maiúsculas.
3. Lembre-se que **"x"**(com aspas) indica o caractere xis, ao passo que **x**(sem aspas) indica o nome de uma variável chamada **x**.
4. Exemplos de dados inválidos: consumo negativo ou tipo de instalação diferente das letras **R**, **I** ou **C**.

Exemplo de Entrada

790.5

C

Exemplo de Saída

Entradas: 790.5 kWh e tipo C

Valor total: R\$ 434.78

```
1 consumo = float(input("1. Digite o consumo total de energia(em kWh): "))
2 tipo = input("2. (R) para residencias || (I) para industrias || (C) para comercios || Tipo: ")
3
4 print("Entradas: ", consumo, " kWh e tipo ", tipo)
5
6 if (consumo > 0):
7     if (tipo == "R"):
8         if (consumo <= 500):
9             valor = round(0.44*consumo, 2)
10        if (consumo > 500):
11            valor = round(0.65*consumo, 2)
12            print("Valor total: R$", valor)
13        elif (tipo == "C"):
14            if (consumo <= 1000):
15                valor = round(0.55*consumo, 2)
16            if (consumo > 1000):
17                valor = round(0.60*consumo, 2)
18                print("Valor total: R$", valor)
19        elif (tipo == "I"):
20            if (consumo <= 5000):
21                valor = round(0.55*consumo, 2)
22            if (consumo > 5000):
23                valor = round(0.60*consumo, 2)
24                print("Valor total: R$", valor)
25        else:
26            print("Dados invalidos")
27    else:
28        print("Dados invalidos")
```

Questão 6

Considere dois números reais a e b , sendo $b > a$. Um número real x pertence ao intervalo $[a, b]$ se $a \leq x \leq b$.

Escreva um programa que leia os números reais x, a, b nesta ordem.

- Se x pertencer ao intervalo, imprima a seguinte mensagem:

`x pertence ao intervalo a, b`

- Caso contrário, imprima a seguinte mensagem:

`x nao pertence ao intervalo a, b`

- Se as entradas forem inválidas, ou seja, se $b \leq a$, imprima a seguinte mensagem:

`Entradas a e b invalidas`

Nas mensagens, substitua as letras `x, a, b` pelos valores fornecidos como entrada.

Dicas

- Note que as três entradas são números reais (`float`).
- Utilize condições encadeadas (`if/elif`).
- Alguns dos operadores booleanos (`and, or, not`) serão úteis.
- Não confunda a vírgula que deve ser impressa (string) com a vírgula que separa um argumento de outro dentro da função `print()`.

Exemplo de Entrada

2.0

1.0

3.0

Exemplo de Saída

2.0 pertence ao intervalo 1.0 , 3.0

```
1 x = float(input("1. Digite um numero x: "))
2 a = float(input("2. Digite o valor de 'a' referente ao intervalo fechado [a,b]: "))
3 b = float(input("3. Digite o valor de 'b' referente ao intervalo fechado [a,b]: "))
4
5 if (b > a):
6     if (x >= a) and (x <= b):
7         print(x, "pertence ao intervalo", a, ",", b)
8     else:
9         print(x, "nao pertence ao intervalo", a, ",", b)
10 else:
11     print("Entradas", a, "e", b, "invalidas")
```

Questão 7

Considere dois intervalos numéricos sobre a reta real: $[a, b]$ e $[c, d]$. Escreva um programa que verifique se existe interseção (pelo menos um ponto em comum) entre os intervalos.

- Se houver **interseção**, o programa deverá imprimir:

Intervalo 1: a, b

Intervalo 2: c, d

Ha intersecao

- Se **não** houver interseção, o programa deverá imprimir:

Intervalo 1: a, b

Intervalo 2: c, d

Nao ha intersecao

- Por fim, se as entradas forem **inválidas**, o programa deverá imprimir:

Intervalo 1: a, b

Intervalo 2: c, d

Entradas invalidas

Nas mensagens, substitua as letras **a, b, c, d** pelos valores fornecidos como entrada.

Dicas

1. Leia os números reais na ordem solicitada: a, b, c, d .
2. Primeiro, verifique se os intervalos são válidos, ou seja, se $b > a$ e $d > c$.
3. Não pressuponha nada com respeito à posição relativa entre os intervalos $[a, b]$ e $[c, d]$. Ou seja, eles podem estar situados antes ou depois um do outro.
4. Antes de programar, teste alguns exemplos numéricos em um rascunho, para encontrar uma regra geral.
5. Alguns dos operadores booleanos (**and**, **or**, **not**) serão úteis.
6. Use letras maiúsculas e minúsculas exatamente conforme o enunciado. **Não use acentos**.

Exemplo de Entrada

4.0

5.0

1.0

2.5

Exemplo de Saída

Intervalo 1: 4.0 , 5.0

Intervalo 2: 1.0 , 2.5

Nao ha intersecao

```
1 a = float(input("1. Digite o valor de 'a' referente ao intervalo fechado [a,b]: "))
2 b = float(input("2. Digite o valor de 'b' referente ao intervalo fechado [a,b]: "))
3
4 c = float(input("1. Digite o valor de 'c' referente ao intervalo fechado [c,d]: "))
5 d = float(input("2. Digite o valor de 'd' referente ao intervalo fechado [c,d]: "))
6
7 print("Intervalo 1:", a, ",", b)
8 print("Intervalo 2:", c, ",", d)
9
10 if (a < b) and (c < d):
11     if ((c >= a) and (c <= b)) or ((d >= a) and (d <= b)):
12         print("Ha intersecao")
13     else:
14         print("Nao ha intersecao")
15 else:
16     print("Entradas invalidas")
```

Questão 8

Escreva um programa leia um **número inteiro**, correspondente ao dia de hoje na semana. Por exemplo, domingo é 0, segunda é 1, terça é 2, ..., sábado é 6.

Se o usuário digitar um número inteiro diferente destes, imprima:

A entrada X eh invalida, onde X é o valor fornecido.

Após isso, peça que o usuário também digite um número de dias no futuro a partir de hoje. Como saída, determine qual é o dia da semana após essa quantidade de dias, com a seguinte mensagem:

Hoje eh X e o dia futuro eh Y

Caso de exemplo 1

```
Entre com o numero do dia de hoje: 1
Entre com o numero de dias apos hoje: 3
Hoje eh segunda e o dia futuro eh quinta
```

Caso de exemplo 2

```
Entre com o numero do dia de hoje: 2
Entre com o numero de dias apos hoje: 8
Hoje eh terca e o dia futuro eh quarta
```

Dicas

1. Atenção para o uso de letras maiúsculas e minúsculas. **Não use acentos.**
2. Alguns dos operadores booleanos (**and**, **or**, **not**) serão úteis.
3. Utilize o operador de resto da divisão (%) para encontrar um número entre 0 e 6.

Exemplo de Entrada

```
4
200
```

Exemplo de Saída

Hoje eh quinta e o dia futuro eh segunda

```

1 dia_h = int(input("1.Digite o numero referente ao dia de hoje na semana: "))
2
3 if (dia_h == 0) or (dia_h == 1) or (dia_h == 2) or (dia_h == 3) or (dia_h == 4) or (dia_h == 5) or (dia_h == 6):
4     dia_f = int(input("2.Digite o numero de dias no futuro a partir de hoje: "))
5     if (dia_f >= 0):
6         # dia atual
7         if (dia_h == 0):
8             msg = "Hoje eh domingo"
9         if (dia_h == 1):
10            msg = "Hoje eh segunda"
11        if (dia_h == 2):
12            msg = "Hoje eh terca"
13        if (dia_h == 3):
14            msg = "Hoje eh quarta"
15        if (dia_h == 4):
16            msg = "Hoje eh quinta"
17        if (dia_h == 5):
18            msg = "Hoje eh sexta"
19        if (dia_h == 6):
20            msg = "Hoje eh sabado"
21
22        # logica para descobrir o dia futuro
23        if ((dia_f % 7) + (dia_h)) % 7 == 0:
24            print(msg + " e o dia futuro eh domingo")
25        if ((dia_f % 7) + (dia_h)) % 7 == 1:
26            print(msg + " e o dia futuro eh segunda")
27        if ((dia_f % 7) + (dia_h)) % 7 == 2:
28            print(msg + " e o dia futuro eh terca")
29        if ((dia_f % 7) + (dia_h)) % 7 == 3:
30            print(msg + " e o dia futuro eh quarta")
31        if ((dia_f % 7) + (dia_h)) % 7 == 4:
32            print(msg + " e o dia futuro eh quinta")
33        if ((dia_f % 7) + (dia_h)) % 7 == 5:
34            print(msg + " e o dia futuro eh sexta")
35        if ((dia_f % 7) + (dia_h)) % 7 == 6:
36            print(msg + " e o dia futuro eh sabado")
37    else:
38        print("A entrada", dia_f, "eh invalida")
39 else:
40     print("A entrada", dia_h, "eh invalida")

```

Questão 9

Escreva um programa que leia 02 valores, x e y , que representam as coordenadas de um ponto no plano cartesiano.

Como saída, determine em que quadrante (Q1, Q2, Q3 ou Q4) o ponto está situado, ou se ele está sobre um dos eixos cartesianos (Eixo X, Eixo Y), ou se ele está na origem $x = y = 0$ (Origem).

Fonte: URI Online Judge | 1041

Dicas

1. Note que o teste com respeito à localização na origem do plano prevalece sobre todos os demais testes.
2. Utilize letras maiúsculas e minúscula **exatamente** como consta no enunciado.
3. Antes de submeter, teste todas as sete possibilidades de saída. **Não se limite ao caso de exemplo.**

Exemplo de Entrada

0

-2.2

Exemplo de Saída

Eixo Y

```
1 x = float(input("1.Digite o valor da coordenada X: "))
2 y = float(input("2.Digite o valor da coordenada Y: "))
3
4 if (x > 0) and (y > 0):
5     print("Q1")
6 if (x < 0) and (y > 0):
7     print("Q2")
8 if (x < 0) and (y < 0):
9     print("Q3")
10 if (x > 0) and (y < 0):
11     print("Q4")
12 if (x == 0) and (y != 0):
13     print("Eixo Y")
14 if (x != 0) and (y == 0):
15     print("Eixo X")
16 if (x == 0) and (y == 0):
17     print("Origem")
```


Questão 10

Escreva um programa que leia um **valor inteiro** m , tal que $1 \leq m \leq 12$.

Como saída, imprima por extenso o **nome do mês** correspondente no ano.

Se a entrada não corresponder a nenhum dos meses do ano, imprima: **numero de mes invalido**

Fonte: URI Online Judge | 1052

Dicas

1. A entrada contém um único valor **inteiro**.
2. Em português, os nomes dos meses são escritos com **inicial minúscula**.
3. **Não se limite ao caso de exemplo.** Antes de submeter, teste exhaustivamente seu programa.

Exemplo de Entrada

1

Exemplo de Saída

janeiro

```
1 mes = int(input("1.Digite o numero do mes: "))
2
3 if (mes >= 1) and (mes <= 12):
4     if (mes == 1):
5         print("janeiro")
6     if (mes == 2):
7         print("fevereiro")
8     if (mes == 3):
9         print("marco")
10    if (mes == 4):
11        print("abril")
12    if (mes == 5):
13        print("maio")
14    if (mes == 6):
15        print("junho")
16    if (mes == 7):
17        print("julho")
18    if (mes == 8):
19        print("agosto")
20    if (mes == 9):
21        print("setembro")
22    if (mes == 10):
23        print("outubro")
24    if (mes == 11):
25        print("novembro")
26    if (mes == 12):
27        print("dezembro")
28 else:
29    print("numero de mes invalido")
```

Questão 11

Uma empresa decidiu dar uma gratificação diferenciada ao melhor funcionário do ano. A gratificação é determinada com base no número de horas extras trabalhadas e no número de horas não trabalhadas, ou seja, que o funcionário faltou ao serviço, conforme tabela a seguir, em que o índice **H** é determinado da seguinte forma:

$$H = (\text{Número de Horas Extras}) - 1/4 * (\text{Número de Horas Não Trabalhadas})$$

Tipo	Índice H	Gratificação
Melhor funcionário	Maior que 400	R\$500,00
Padrão	Até 400	R\$100,00

Escreva um programa que leia:

1. O **número de horas** extras;
2. O **número de horas** que o funcionário faltou.

Considere as horas informadas como sendo um número real, por exemplo 3.5 horas.

Como saída, imprima a seguinte mensagem:

E extras e F de falta

R\$ G

Nas mensagens, substitua a letra G pelo valor correspondente.

Dicas

1. Os valores em moeda devem ser arredondados em **duas casas**.
2. A ordem de leitura das variáveis de entrada deve obedecer àquela explicitada no enunciado.
3. Considere que o usuário sempre fornece entradas válidas, ou seja, números reais positivos.
4. Atenção para o uso de caracteres maiúsculos e minúsculos. Não use acentos, para evitar erros.

Exemplo de Entrada

10.5

2.0

Exemplo de Saída

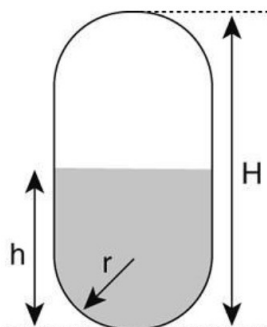
10.5 extras e 2.0 de falta

R\$ 100.0

```
1 ext = float(input("1.Digite o numero de horas extras trabalhadas: "))
2 falt = float(input("2.Digite o numero de horas que faltou: "))
3
4 h = (ext) - (1/4)*falt
5
6 if (h <= 400):
7     print(ext, "extras e", falt, "de falta")
8     print("R$ 100.0")
9 else:
10    print(ext, "extras e", falt, "de falta")
11    print("R$ 500.0")
```

Questão 12

Um tanque de combustível tem o seguinte formato:



Escreva um programa que leia as seguintes medidas (em metros), nesta ordem:

1. H : altura total do tanque
2. h : nível de combustível no tanque
3. r : raio dos bojos semiesféricos inferior e superior

Como saída, calcule o **volume** de combustível (em litros), com até **três** casas decimais de precisão.

- Se as entradas forem válidas, imprima como saída:

Entradas: H , h , r

Volume: V litros

- Se as entradas **não** forem válidas, imprima como saída:

Entradas: H , h , r

Entradas invalidas

Nas mensagens, substitua as letras H , h , r , V pelos valores correspondentes.

Validação dos dados

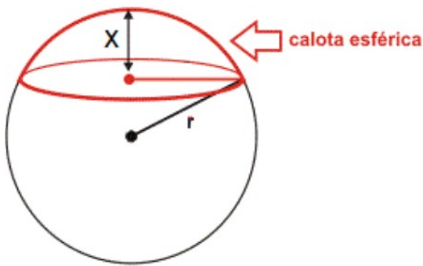
Verifique se as entradas fornecidas são válidas, ou seja:

- Todas as três entradas devem ser maiores que zero; e
- H deve ser maior que h e maior que $2.r$

Fórmulas úteis (adapte-as ao contexto)

1. Volume do **cilindro** de raio r e altura x : $V = \pi r^2 x$
2. Volume da **esfera** de raio r : $V = \frac{4}{3} \pi r^3$

3. Volume da **calota** esférica de raio r e altura x : $V = \frac{\pi}{3}x^2(3r - x)$



Dicas

1. Lembre-se que 1 m³ equivale a 1000 litros.
2. O nível de combustível h pode assumir **três** posições em relação à altura do tanque H e o raio r . Cada posição corresponde a uma condição a ser testada no **if** ou **elif**:
 - a. Semiesfera inferior.
 - b. Cilindro intermediário.
 - c. Semiesfera superior.
3. Se o nível de combustível h estiver na semiesfera **superior**, então a maneira mais fácil de encontrar o volume de combustível é subtraindo o **volume de ar** do volume total do tanque.
4. Adapte as variáveis usadas nas fórmulas para as variáveis que você está usando no programa.
5. Teste seu programa para as diversas possibilidades das entradas H, h, r .

Exemplo de Entrada

2.0

1.5

0.5

Exemplo de Saída

Entradas: 2.0 , 1.5 , 0.5

Volume: 1047.198 litros

```

1 from math import *
2
3 alt_t = float(input("1.Digite a altura total do tanque(metros): "))
4 niv = float(input("2.Digite o nivel de combustivel no tanque(metros): "))
5 raio = float(input("3.Digite o valor do raio dos bojos semiesfericos(metros): "))
6
7 print("Entradas:", alt_t, ",", niv, ",", raio)
8
9 if (alt_t > 0) and (niv > 0) and (raio > 0) and (alt_t > niv) and (alt_t > (2*raio)):
10     if (niv <= raio): # semiesfera inferior
11         vol = (pi/3) * (niv**2) * (3*raio - niv)
12     if (niv > raio) and (niv <= (alt_t - raio)): # dentro do cilindro
13         vol = (4/3) * pi * (raio**3)
14         vol = (vol/2) + (pi * (raio**2) * (niv - raio))
15     if (niv > raio) and (niv > (alt_t - raio)): # semiesfera superior
16         vol = (4/3) * pi * (raio**3)
17         vol = vol + (pi * (raio**2) * (alt_t - 2*raio))
18         vol = vol - ((pi/3) * (alt_t - niv)**2 * (3*raio - (alt_t - niv)))
19     vol = round(vol*1000, 3)
20     print("Volume:", vol, "litros")
21 else:
22     print("Entradas invalidas")

```