

Prova

Leia atentamente as instruções a seguir. Elas fazem parte da prova:

- Desligue seu celular e guarde em sua bolsa.
- Salve seu trabalho frequentemente.
- Ao término, salve e submeta este arquivo ao tidia na atividade Prova, da mesma forma como é feito nas listas de exercícios.
- A compreensão e interpretação do enunciado é parte integrante da avaliação.
- Não é permitida a consulta a nenhum material online (Internet) nem offline (cadernos, livros, etc.) e nem a colegas.
- Em caso de fraude, **todos** os envolvidos receberão **nota final zero** na disciplina, sem prejuízo de outras sanções.

Critério de correção de cada exercício: 0 se incorreto e/ou não executa; 0.5 se correto e executa mas não segue os padrões de submissão (**leia com atenção como deve ser a saída: nenhum outro texto deve ser impresso pelo seu programa além do que estiver especificado no campo "Saída"**); 1 caso contrário. Seu programa está **correto** se funciona para qualquer entrada especificada no campo "Entrada" do problema.

BOA PROVA!

Questão 1 (Valor: 2,0)

Faça um programa que leia os valores (decimais) correspondentes aos três lados a , b e c de um triângulo. O programa deve então calcular a área A do triângulo utilizando a fórmula de Heron dada a seguir:

$$A = \sqrt{s(s - a)(s - b)(s - c)}$$

onde

$$s = \frac{a + b + c}{2}$$

Entrada: O programa deve receber três números decimais positivos que formam um triângulo.

Saída: A resposta consiste de um único número decimal, que indica a área do triângulo dado.

Exemplos de ENTRADA	SAÍDA esperada (valor aproximado)
3 4 5	6.0
15 20.3 7.8	49.25
3.14 2.71 4.2	4.25
42 42 42	763.83

```
In [2]: a = float(input("Digite o valor do primeiro lado: "))
b = float(input("Digite o valor do segundo lado: "))
c = float(input("Digite o valor do terceiro lado: "))

s = (a + b + c)/2
A = (s * (s-a) * (s-b) * (s-c))**0.5

print(A)
```

Digite o valor do primeiro lado: 15
Digite o valor do segundo lado: 20.3
Digite o valor do terceiro lado: 7.8
49.25503622727315

Questão 2 (Valor: 2,0)

Leia quatro números (N_1, N_2, N_3, N_4), que correspondem às quatro notas de um aluno. Calcule a média com pesos 2, 3, 4 e 1 para cada uma destas notas, respectivamente, e mostre esta média acompanhada pela mensagem `Media:` . Se esta média for maior ou igual a 7.0, imprima também a mensagem `Aluno aprovado.` . Se a média calculada for inferior a 5.0, imprima a mensagem `Aluno reprovado.` . Se a média calculada for um valor entre 5.0 e 6.9, inclusive estes, o programa deve imprimir a mensagem `Aluno em exame.` . Cada mensagem deve ser impressa em uma única linha (veja os exemplos).

No caso do aluno estar em exame (e apenas nesse caso), leia ainda um valor correspondente à nota do exame obtida pelo aluno. Imprima então a mensagem `Nota do exame:` acompanhada pela nota digitada. Recalcule a média (some a pontuação do exame com a média anteriormente calculada e divida por 2) e imprima a mensagem `Aluno aprovado.` , caso a média final seja 5.0 ou mais, ou `Aluno reprovado.` , caso a média tenha ficado menor do que 5.0. Para estes dois casos (aprovado ou reprovado após ter pego exame), apresente na última linha uma mensagem `Media final:` seguida da média final para esse aluno.

Entrada: A entrada inicialmente consiste de quatro números reais N_1, N_2, N_3, N_4 indicando as quatro notas de um aluno. Ela pode ser seguida de um quinto número real que indica a nota obtida pelo aluno no exame.

Saída: As mensagens devem ser impressas conforme a descrição do problema. Não esqueça de imprimir uma mensagem por linha apenas.

Exemplo de entrada	Saída esperada
2.0 4.0 7.5 8.0	Media: 5.4
6.4	Aluno em exame.
	Nota do exame: 6.4
	Aluno aprovado.
	Media final: 5.9
Exemplo de entrada	Saída esperada
2.0 6.5 4.0 9.0	Media: 4.8
	Aluno reprovado.
Exemplo de entrada	Saída esperada
9.0 4.0 8.5 9.0	Media: 7.3
	Aluno aprovado.

```
In [2]: n1 = float(input("Digite a nota da primeira prova: "))
n2 = float(input("Digite a nota da segunda prova: "))
n3 = float(input("Digite a nota da terceira prova: "))
n4 = float(input("Digite a nota da quarta prova: "))

# numa média ponderada, dividimos pela soma dos pesos, que no caso é 10
media = (2*n1 + 3*n2 + 4*n3 + 1*n4)/10
print("Media:", media)

if media >= 7:
    print("Aluno aprovado.")
elif media < 5:
    print("Aluno reprovado.")
else:
    print("Aluno em exame.")
    exame = float(input("Digite a nota do exame: "))
    media = (media + exame)/2
    if media >= 5:
        print("Aluno aprovado.")
    else:
        print("Aluno reprovado.")
    print("Media final:", media)
```

Digite a nota da primeira prova: 9
Digite a nota da segunda prova: 4
Digite a nota da terceira prova: 8.5
Digite a nota da quarta prova: 9
Media: 7.3
Aluno aprovado.

Questão 3 (Valor: 3.0)

Um número inteiro positivo é dito perfeito se a soma dos seus divisores positivos (excluindo ele mesmo) é igual ao próprio número. Por exemplo, 6 é perfeito pois $1 + 2 + 3 = 6$. Faça um programa que determine se um dado número é perfeito. Lembre-se que os divisores de um número n são todos os números entre 1 e n que dividem n de forma exata (sem resto).

Entrada: O programa deve receber um único número inteiro n , com $n \geq 1$.

Saída: A resposta consistirá de uma única linha, contendo a palavra Sim , se n for perfeito, e Não , caso contrário.

```
In [5]: n = int(input("Digite um número inteiro maior que 1: "))

soma_div = 0
# vamos testar todos os candidatos a divisores do n: os números entre 1 e n-1
for i in range(1,n):
    # se o número for divisor do n, então some este à variável soma_div
    if n % i == 0:
        soma_div = soma_div + i

# se o que foi acumulado na soma for igual ao número, temos um número perfeito
if soma_div == n:
    print("Sim")
else:
    print("Não")
```

Digite um número inteiro maior que 1: 28
Sim

Questão 4 (Valor: 3.0)

Considere o seguinte algoritmo:

- (1) receba um número n
- (2) se n for 1, então pare
- (3) se n for ímpar, então faça $n = 3*n+1$
- (4) senão, faça $n = n/2$
- (5) vá para a linha (2)

Por exemplo, quando $n = 22$ é lido, a seguinte sequência de números será gerada: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1 .

Há uma conjectura de que o algoritmo acima irá terminar (ou seja, o número 1 será gerado) para qualquer valor n positivo. Desconsiderando a simplicidade do algoritmo, não é sabido se a conjectura é verdadeira. Ela foi verificada, no entanto, para todos os inteiros n tais que $0 < n < 1.000.000$ (e, de fato, para muito mais números maiores que isto).

Dado um valor n , é possível determinar a quantidade de números gerados na sequência (incluindo o 1). Para um dado n , isto é chamado *tamanho cíclico* de n . No exemplo anterior, o tamanho cíclico de 22 é 16.

Para quaisquer dois números i e j , você deve determinar o tamanho cíclico de todos os números entre i e j inclusive ambos (ou seja, para todo n tal que $i \leq n \leq j$).

Entrada: O programa deve receber dois números inteiros i e j , com $0 < i \leq j \leq 10.000$.

Saída: A resposta consistirá de $j - i + 1$ linhas, uma para cada número n entre i e j (inclusive). Cada linha deve conter dois números: o n e o tamanho cíclico de n .

Exemplo de entrada	Saída esperada
10 22	10 7
	11 15
	12 10
	13 10
	14 18
	15 18
	16 5
	17 13
	18 21
	19 21
	20 8
	21 8
	22 16

```
In [7]: i = int(input("Digite o primeiro número do intervalo: "))
j = int(input("Digite o segundo número do intervalo: "))

# para todo valor dentro do intervalo de i a j (inclusive) precisamos calcular o tamanho cíclico
for n in range(i,j+1):
    tam_ciclico = 1 # variável para contar o tamanho cíclico
    auxn = n # salvando o valor de n para imprimir depois
    # precisamos simular a criação da sequência para poder contar o tamanho dela
    while n > 1:
        if n % 2 == 0:
            n = n / 2
        else:
            n = 3*n + 1
        # contamos um número a mais toda vez que o geramos
        tam_ciclico = tam_ciclico + 1
    print(auxn, tam_ciclico)
```

Digite o primeiro número do intervalo: 10

Digite o segundo número do intervalo: 22

10 7

11 15

12 10

13 10

14 18

15 18

16 5

17 13

18 21

19 21

20 8

21 8

22 16

In []: