

## **Lista 4 - Condicionais/Recursão**

**Submissão:** Crie um programa para cada questão abaixo, ou seja um executável como mostrado em aula, depois compacte eles em um arquivo .zip(só aceitarei essa forma de compactação) e submeta ao sigaa.

**Exemplo:** Aluno de matrícula 000000 deve criar os arquivos: 000000\_q1.py, 000000\_q2.py, 000000\_q3.py, 000000\_q4.py e 000000\_q5.py. Deve compactar usando a técnica zip e submeter ao sistema.

### **1. (0,05) Função como argumento de outra recursiva.**

Escreva um programa que leia dois inteiros: a e b. Este programa deve passar a e b como argumentos da função somatório que, por sua vez, irá realizar o somatório dos resultados da chamada a uma função cubo, que deve ser passada como parâmetro. Funções em python podem ser passadas como argumentos para outras, veja o exemplo abaixo.

```

>>> def quadrado(n: int) -> int:
        return n*n

>>> quadrado
<function quadrado at 0x7fe2ac0aeaf0>
>>> def teste(n: int, f: "função") -> int:
        return f(n)

>>> teste(2, quadrado)
4

```

A função cubo é simples de implementar, basta pegar seu parâmetro e elevar ao cubo. Já a função somatório deve ser implementada de forma recursiva e retorna o resultado da soma dos cubos dos números no intervalo [a,b], a e b incluídos.

## Entrada e saída

O programa deve ler dois inteiros, um por linha, e fornecer a saída esperada.

Entrada de exemplo	Saída de exemplo
1 5	225
6 7	559
0 1	1

Como exemplo veja a primeira sequência a ser calculada:

Argumentos: 1 e 5; Retorno: 225

Operação:  $1^2 + 2^2 + 3^2 + 4^2 + 5^2$

## Observações:

Implementação recursiva da função somatório.

Teste usando casos extremos, ou seja, que são válidos mas incomuns.

## 2. (0,05) Conjectura de Collatz.

Considere o seguinte procedimento em um número inteiro positivo

- se  $n$  é par, divida por 2;
- se  $n$  é ímpar, multiplique por 3, some 1 e divida por 2.

A conjectura de Collatz diz que se repetirmos esse procedimento no resultado da execução anterior um número suficiente de vezes, eventualmente vamos chegar ao número 1. Faça uma função que dado um número inteiro positivo, determina o número de repetições do procedimento para chegar em 1.

### Entrada e saída

Entrada: um número inteiro positivo  $n$ . Saída: O número de passos para convergir para 1.

Entrada de exemplo	Saída de exemplo
15	12

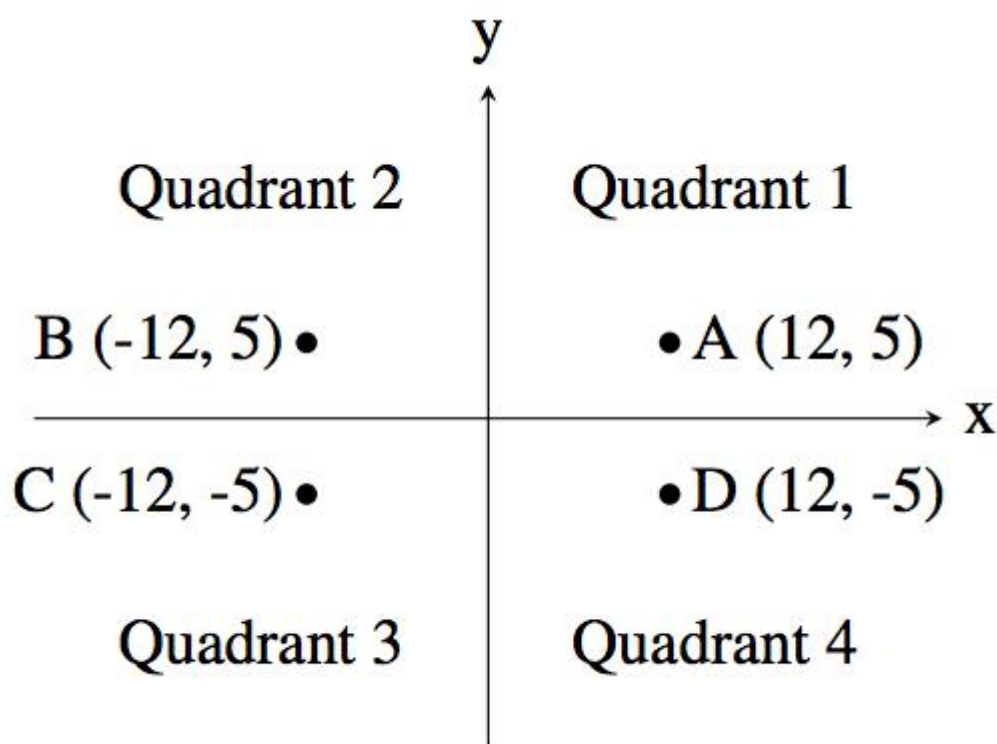
## Observações:

Você deve usar recursão.

Teste usando casos extremos, ou seja, que são válidos mas incomuns.

### 3. (0,005) Seleção do quadrante

Um problema comum em matemática é determinar em qual quadrante um determinado ponto se encontra. Existem quatro quadrantes, numerados de 1 a 4, conforme mostrado no diagrama abaixo:



Por exemplo, o ponto A, que está nas coordenadas (12, 5), encontra-se no quadrante 1, pois ambos os valores x e y são positivos, e o ponto B está no quadrante 2, pois seu valor x é negativo e seu valor y é positivo.

Seu trabalho é pegar um ponto e determinar em que quadrante ele está. Você pode presumir que nenhuma das duas coordenadas será 0.

## Entrada e saída

Entrada:

A primeira linha de entrada contém o inteiro  $x$  ( $-1000 \leq x \leq 1000$ ;  $x \neq 0$ ). A segunda linha de entrada contém o inteiro  $y$  ( $-1000 \leq y \leq 1000$ ;  $y \neq 0$ ).

Saída:

Mostre o número do quadrante (1, 2, 3 ou 4) para o ponto  $(x, y)$ .

Entradas de exemplo	Saída de exemplo
10 6	1
9 -13	4

## Observações

Teste usando casos extremos, ou seja, que são válidos mas incomuns.

### 4. (0,005) Saudações

Agora que o Snapchat e o Slingshot são muuuuito de 2018, os adolescentes de todo o mundo mudaram para o novo app chamado BAPC (comunicação bidirecional e privada). Este aplicativo tem algumas regras sociais mais rígidas do que os anteriores. Por exemplo, se alguém se despedir usando **Later!**, espera-se que a outra pessoa responda com **Alligator!**. Você não pode acompanhar todas essas convenções sociais e decidir automatizar as respostas necessárias, começando pela mais importante: as saudações.

Quando seu parceiro de conversação abre com he...ey, você tem que responder hee...eey também, mas usando o dobro de e's!

## **Entrada e Saída**

A entrada consiste em uma linha contendo uma única string *s* conforme especificado, com comprimento de no mínimo 3 e no máximo 1000.

A saída é a requisitada das convenções acima.

Exemplo de entrada	Exemplo de saída
hey	heey
Later!	Alligator!
Hello	Hello

## **Observações**

Não é para usar repetição.

Procure um método de string capaz de resolver essa adição de caracteres 'e'.

## **5. (0,005) Parque de estacionamento**

Considere um parque de estacionamento que pratica os preços seguintes:

- 1ª hora: R\$2;
- 2ª hora: R\$1,5;
- a partir da 2ª hora: R\$1 por hora;

O tempo de permanência no parque é contabilizado em horas e minutos. Por exemplo, se um carro

permanecer 2 horas e 30 minutos no parque,  
pagará R\$2 (1ª hora) + R\$1,5 (2ª hora) + R\$0,5 (30 minutos a R\$1/hora) = R\$4.

## **Entrada e Saída**

Elabore um algoritmo que, lido o tempo que determinado veículo permaneceu estacionado no parque, diga a quantia que deve ser paga.

## **Observações**

Teste os casos extremos.