

Segundo Trabalho - Computational Thinking

Regras:

- Este trabalho pode ser feito por no **máximo** três pessoas, não haverá nenhuma exceção.
- Serão aceitos trabalhos individuais e em dupla.
- Date de entrega: 27/05/2020 através da área de trabalhos do portal do aluno.
- Como artefatos da entrega, você deverá fazer o upload de um arquivo zip contendo os fontes em Python e o nome dos integrantes do trabalho.
- Para os exercícios abaixo não será permitido o uso de nenhuma biblioteca que manipula as datas. Todos cálculos deverão ser feitos usando os operadores aritméticos, condicionais, comandos de decisão e repetição que estudamos.
- Para os exercícios envolvendo string, use apenas os conceitos que foram passados em sala de aula.

Para manipular tuplas

Considere o seguinte código em Python:

```

1 def calcula(a, b):
2     return a // b, a % b
3
4 res = calcula(5, 3)
5 print(res)
6 #para acessar individualmente o elemento da tupla, fazemos:
7 print(res[0])
8 print(res[1])

```

Caso você precise acessar o valor armazenado em uma tupla, use como exemplo o código acima fazendo as devidas modificações.

Dicas para fazer os exercícios de datas

Observações: considere que qualquer data digitada obedece ao calendário gregoriano, que é o calendário atual. Você também pode considerar que todas as datas que forem passadas como parâmetros são datas existentes, ou seja, não é necessário validar elas.

Dica, para fazer esses exercícios, crie funções auxiliares. Por exemplo, comece criando uma função que recebe uma data, representada pelos três números inteiros, e retorna a data de amanhã.

Questões

1. Uma data pode ser representada, de maneira bem simples, utilizando três números inteiros: dia, mes e ano. Sua tarefa neste exercício, é criar uma função que recebe como parâmetros três numeros inteiros representando uma data e um outro número inteiro representando uma

quantidade de dias. Essa quantidade de dias, deverá ser somada a essa data retornando uma data futura. Por exemplo, suponha o dia 13/05/2020 e 25 dias que serão somados, sua função deverá retornar 08/06/2020. Retorne o resultado da função como uma tupla.

2. Crie uma nova função que ao invés de somar os dias sobre uma data, irá subtrair eles. Por exemplo, suponha o dia 13/05/2020 e 7 dias que serão debitados, sua função deverá retornar a data 06/05/2020.
3. Ainda sobre datas, escreva uma função que recebe duas datas (cada uma das sendo representada por 3 números inteiros) e retorna a quantidade de dias existentes entre as duas datas. Por exemplo, você poderia informar a data de seu aniversário e a data de hoje, sua função retornará a quantidade de dias que você viveu.

Observações: considere que qualquer data digitada obedece ao calendário gregoriano, que é o calendário atual. Você também pode considerar que todas as datas que forem passadas como parâmetros são datas existentes, ou seja, não é necessário validar elas.

Dica, para fazer esses exercícios, crie funções auxiliares. Por exemplo, comece criando uma função que recebe uma data, representada pelos três números inteiros, e retorna a data de amanhã.

4. Escreva uma função em Python que recebe como parâmetros uma String e retorna a quantidade de vogais existentes nessa String.
5. Escreva uma função em Python que recebe como parâmetro uma String e retorna uma outra String invertida. Por exemplo, suponha que seja passado como parâmetro a String "Marrocos", sua função deverá retornar a String "socorraM".

Boa sorte!

Eduardo Gondo