

**Lista de Exercícios Avaliativa sobre comando de repetição e *strings*.**

**Desenvolva os programas solicitados nas questões respeitando as seguintes regras:**

- a) use a linguagem de programação Python (versão 3.X);
- b) **não empregue conhecimentos ainda não apresentados.** Assim:
  - se a atividade trata até comando condicional:  
não use repetição, listas, *comprehensions*, funções, operações complexas sobre *strings*, entre outros;
  - se a a atividade trata até comandos de repetição:  
não use listas, *comprehensions*, funções, operações complexas sobre *strings*, entre outros;
  - e assim por diante;
- c) **não empregue bibliotecas ou funções prontas do Python**, exceto quando o enunciado da questão definir essa possibilidade;
- d) **Comente o código;**
- e) **Trate os erros de entrada inválida;**
- f) desenvolva a atividade com até mais um colega; **apenas um dos membros da dupla entrega a atividade**; se mais de um entregar apenas o primeiro será considerado na avaliação;

**Para a entrega siga as seguintes regras:**

- a) Crie um repositório no [www.github.com](https://www.github.com) com o nome ProgComp-2024.2
- b) Dentro do repositório deverá ter uma pasta com o nome Unidade01-Avaliacao02 com os programas que respondem às questões;
- c) Dentro da pasta crie os programas de resposta às questões com os: q1.py, q2.py e assim sucessivamente;
- d) A resposta a essa atividade deve um pequeno texto contendo apenas:
  - Os nomes e matrículas dos membros grupo
  - O nome do usuário [www.github.com](https://www.github.com) que está entregando a atividade.

**A avaliação será feita tendo por base o nome do usuário, com o nome do repositório e pasta acima citados.**

- e) A correção será realizada com o último *commit* anterior à data de entrega. Também pode ser objeto de avaliação a frequência em que cada membro do grupo fez *commits*;

**Qualquer infração às regras acima implica na não correção da atividade e/ou questão, com a nota sendo zerada para o item/atividade.**

- 1) (2 pontos) Um número é dito decrescente quando os seus dígitos ao serem observados em pares sucessivos, nenhum par deles tem o dígito da direita maior do que o da esquerda. Ex: 954331 é um número decrescente (3 não é menor do que 1; 3 não é menor do que 3; 4 não é menor do que 3; 5 não é menor do que 4; 9 não é menor do que 5;). Nos números crescentes, por sua vez, dígitos sucessivos têm o da esquerda nunca maior do que o da direita; é o que ocorre em 46779. Faça um programa que contar quantos números decrescentes existem de 10 até 987631 (inclusive).
- 2) (2 pontos) Faça um programa que conta quantos números palíndromos existem entre 10 e 100000. Um número é palíndromo quando tem o mesmo valor se lido da esquerda para a direita ou da direita para a esquerda. Ex: 98189 é palíndromo. Não use *strings* na sua resposta.
- 3) (2 pontos) Faça um programa para encontrar quantos pares de números primos, menores do que 1000000, existem que são ímpares consecutivos.  
Exemplo: Se considerarmos que o limite é 50, existem 6 pares:  
(3, 5), (5, 7), (11, 13), (17, 19), (29, 31), (41, 43)
- 4) (2 pontos) Faça um programa que implementa o termo ([www.term.ooo](http://www.term.ooo)) na sua versão dueto. Nessa versão existem duas palavras a serem descobertas. Em cada rodada o usuário informa uma palavra, que é confrontada com as duas a serem descobertas. O usuário vence quando descobre as duas palavras em até sete tentativas. Mostre as palavras com as mesmas cores do referido sítio e somente aceite como entrada as palavras listadas no arquivo que acompanha essa atividade.
- 5) (2 pontos) Faça um programa que conta quantos dias já se passaram desde 27/04/1968 (considere esse dia como o dia zero)? Se aquele dia caiu num sábado, quantos outros sábados já existiram depois dele até hoje?

```
import datetime
date = datetime.datetime.today()
dia = date.day
mes = date.month
year = date.year
```

Boa sorte.