

Aluno(a): _____

-
- ❖ A prova é individual e contém **DUAS QUESTÕES (VEJAM AS DUAS PÁGINAS)**.
 - ❖ **Caso seja identificada cópia, nota zero será atribuída a todos os envolvidos.**
 - ❖ Respostas devem ser entregues EXCLUSIVAMENTE no TEAMS. **Só será aceita como entrega válida os códigos em ARQUIVOS .py.** Anexos .ipynb, arquivos sem extensão, ou com qualquer outra extensão diferente de .py serão desconsiderados
-

1. [6,0 pontos] Utilizando a linguagem Python, defina:

a. uma classe `ConjuntoDeInteiros`, que possuirá:

- i. um `__init__` com o parâmetro opcional "capacidade de armazenamento" deste conjunto, que cria um conjunto vazio e definir a capacidade máxima especificada. O valor padrão do parâmetro deve ser 100.
- ii. um método inserção, que recebe como parâmetro um inteiro, e o adiciona no conjunto (caso ainda haja espaço disponível)
- iii. um método união que recebe como parâmetro uma instância da classe `ConjuntoDeInteiros` e retorna uma nova instância desta classe contendo a união da instância encapsulada e a instância recebida como parâmetro
- iv. um método interseção que recebe como parâmetro uma instância da classe `ConjuntoDeInteiros` e retorna uma nova instância desta classe contendo a interseção da instância encapsulada e a instância recebida como parâmetro.
- v. um método diferença que recebe como parâmetro uma instância da classe `ConjuntoDeInteiros` e retorna uma nova instância desta classe contendo a diferença entre a instância encapsulada e a instância recebida como parâmetro. Esta operação deve garantir que os dois conjuntos possuem a mesma quantidade de elementos.

Dica: nos itens *iii*, *iv* e *v*, usem *list comprehension*, *map*

- vi. um método `__str__`. Por exemplo, para um conjunto com valores de 1 a 3, o método deve gerar: "`(1, 2, 3)`"

b. Um arquivo principal `main.py` que:

- i. instancia dois conjuntos de inteiros e insere aleatoriamente 50 valores entre 1 e 100 em cada um dos dois conjuntos.

Aluno(a): _____

- ii. **exibe** na tela e **salva** num arquivo de texto `"conjuntos.txt"` os dois conjuntos gerados, assim como a união, a interseção e a diferença entre estes conjuntos.

2. [4,0 pontos] Utilizando a linguagem `Python`, implemente uma classe `Hora` que:

- possua um único atributo que armazene a quantidade de segundos a partir da meia-noite;
- possua um `__init__` que cria e define quantidade de segundos a partir dos três parâmetros que recebe: horas, minutos e segundos;
- os métodos `get_horas`, `get_minutos` e `get_segundos`;

Exemplos:

quantidade de 300 segundos -> 00h 05m 00s

quantidade de 366 segundos -> 00h 06m 06s

- o método `add_horas` que recebe uma outra instância da classe `Hora` e devolve uma nova instância desta classe que tenha as horas, minutos e segundos das duas instâncias somados;
- o método `__str__` que devolve uma string no formato `hh:mm:ss`.

Sugestão: utilizem os `gets` implementados anteriormente

Atenção para a necessidade de conversão nos b. e c.!

OBSERVAÇÕES GERAIS:

- Façam todas as validações necessárias (identifique-as).
- Assumam a técnicas de encapsulamento de dados para todas as questões.
- Teste seu código antes de submeter.

Boa Prova!