

## Trabalho 1 da disciplina de Algoritmos e Lógica de Programação

**Obs: esse trabalho deverá ser desenvolvido individualmente**

Responda as questões a seguir:

**Questão 1\_2,5 pontos)** Considere a necessidade de elaborar programas em Python em que foi solicitada ao programador a utilização de operadores de atribuição simplificada, tais como: +=, -=, \*=, /=, %=. Dessa forma, faça uma explicação sobre os operadores de atribuição (usando um documento no Word) além de criar um enunciado e um programa em Python onde será necessário usar as expressões a seguir:

- a)  $x += y$
- b)  $x -= y$
- c)  $x *= y$
- d)  $x /= y$
- e)  $x \% = y$
- f) soma += 10;
- g) sub -= 5;

**Obs:** resolver essa questão em um documento do Word (salvar o arquivo com o nome: T1\_Q1.doc) e uma IDE (Python) para criar um programa com extensão \*.py (salvar o arquivo com o nome: T1\_Q1.py)

**Questão 2\_2,0 pontos)** Escreva um programa em Python que permita a leitura de um número inteiro N do intervalo de 10 .. 99. Em seguida o programa deve calcular e imprimir um ÚNICO número inteiro tendo a seguinte estrutura DSU, onde D é o dígito da dezena do número original, U é o dígito da unidade do número original e S é a soma dos dígitos da dezena e unidade. Veja os exemplos:

Para o valor 13 deverá imprimir 143

Para o valor 55 deverá imprimir 5105

Para o valor 99 deverá imprimir 9189

**Obs:** assim, para um valor de entrada válido seu programa deve exibir as mensagens abaixo.

```
Digite um valor para N: 99
Dezena: 9
Unidade: 9
Soma da dezena com unidade: 18
Valor total final: 9189
```

Caso contrário imprimir: "valor de entrada nao tem 2 digitos"

```
Digite um valor para N: 777
Valor de entrada nao possui 2 digitos
```

**Obs:** resolver essa questão usando uma IDE (Python) e salvar o arquivo com o nome: T1\_Q2.py

**Questão 3\_2,5 pontos)** Criar um programa em Python para ler dois números inteiros positivos. O primeiro número deve ser armazenado em A e deve ser um número de 4 dígitos, já o segundo número deve ser armazenado em B e deve ser um número de apenas 2 dígitos. Em seguida o programa deve verificar se B corresponde aos últimos dígitos de A, se sim imprimir a mensagem “encaixa”, caso contrário “não encaixa”. Para valores que não possuam 4 dígitos e 2 dígitos, para A e B respectivamente, imprimir a mensagem “Dados de entrada inadequados”.

Exemplos:	A	B	Mensagem
	5478	78	“Encaixa”
	2367	36	“Não Encaixa”
	1234	34	“Encaixa”
	345	45	“Dados de entrada inadequados”

Obs: resolver essa questão usando uma IDE (Python) e salvar o arquivo com o nome: T1\_Q3.py

**Questão 4\_3,0 pontos)** Cite um problema (situação) da vida real em que você precisa usar o conceito estruturas condicionais aninhadas, assim como a resolução de expressões aritméticas baseadas no uso de operadores de atribuição simplificada. Dessa forma, elabore um enunciado e respectivo programa em Python que use estruturas condicionais aninhadas e as expressões mencionadas por você. O programa em Python deve ser capaz de solucionar o problema proposto.

Obs1: resolver essa questão usando uma IDE (Python), adicionando o problema (situação) e enunciado elaborado dentro do programa (use comandos de comentário “”). Salve o arquivo com o nome: T1\_Q4.py

Obs2: essa questão será pontuada de 0 a 3 considerando criatividade, singularidade, nível de dificuldade atribuído ao problema (situação) e código fonte (corretude e legibilidade).

### Observação Final

Os arquivos: T1\_Q1.docx, T1\_Q1.py, T1\_Q2.py, T1\_Q3.py e T1\_Q4.py deverão ser adicionados em um arquivo zip denominado SobrenomeNome\_Trabalho1.zip e enviado a partir da opção Tarefas do Microsoft Teams da equipe associada a disciplina de Algoritmos e Lógica de Programação.

Trabalhos enviados por e-mail não serão considerados.