

### Atividade Prática 6 - Algoritmos e Programação

1. Dado uma sequência de números inteiros em uma única linha, determinar a quantidade de vezes que cada um deles ocorre na própria sequência.

Sequência: -1 3 0 5 0 5 -1 2 -5 0

Saída:

-1 ocorre 2 vez(es)  
3 ocorre 1 vez(es)  
0 ocorre 3 vez(es)  
5 ocorre 2 vez(es)  
2 ocorre 1 vez(es)  
-5 ocorre 1 vez(es)

Você não deve escrever a saída mais de uma vez para o mesmo valor. Dica: utilize uma lista auxiliar para armazenar para quais valores você já escreveu a saída.

2. [MEMÓRIA] Considere o código abaixo:

```
1. n = int(input())
2. v = []

3. v.append(10)
4. v.append(15)

5. i = 0
6. while i < n:
7.     v.append(float(input()))
8.     i += 1
```

Considere também o seguinte contexto:

- Cada posição de memória ocupa 1 byte;
- O processador em questão utiliza endereços de memória de 16 bits (2 bytes);
- Um valor inteiro ocupa 32 bits (4 bytes);
- Um valor float ocupa 64 bits (8 bytes).
- Todas as regiões de memória podem ser utilizadas, exceto as posições destacadas em vermelho;
- Qualquer vetor no Python recebe, inicialmente, 8 bytes de memória para armazenar valores. Assuma que o sistema operacional decidiu alocar o vetor, inicialmente, na posição 8 de memória;
- Escolha a região de memória de sua preferência para alocar a variável `n` na memória;
- Você pode realocar as variáveis do seu programa na memória sempre que houver necessidade de mais espaço.

Utilize a tabela disponibilizada neste link para representar qual será o estado da memória em dois momentos diferentes da execução:

- Logo após a execução da linha 4;
- Logo após a execução da linha 8.