

Roteiro de Perguntas – Estruturas de Dados em Python

2. Dicionários em Python

- O que acontece se você tentar acessar uma chave que não existe?
- É possível adicionar uma nova chave ao dicionário depois de criado?

3. Tuplas

- O que acontece se você tentar alterar um valor de uma tupla?
- Quando seria melhor usar uma tupla em vez de uma lista?
- Como se acessa os elementos de uma tupla?

4. Arrays NumPy

- Qual é o tipo de dados armazenado em um array? Como verificar isso?
- Todos os elementos têm o mesmo tipo?
- Qual a diferença entre `arange()` e `linspace()`?
- Para que serve a função `reshape()`?

5. Performance: lista vs array

- Qual estrutura foi mais rápida? Por quê?

6. Operadores e broadcasting

- O que acontece se os arrays tiverem tamanhos diferentes?
- A operação foi feita elemento a elemento? Como você sabe?

7. ufuncs (funções universais)

- O que a função faz exatamente?
- A função altera o array original ou retorna um novo?

8. Indexação e slicing

- O que representa o índice negativo?
- O slicing inclui o índice final?
- O resultado é uma cópia ou uma view?

9. View (shallow copy) vs cópia profunda

- Qual é o `id()` do array original e da cópia? Eles são iguais?

- O que acontece ao alterar a view? Isso afeta o original?
- Quando devemos usar cópia profunda?

10. Flatten vs Ravel

- Ambos retornam arrays unidimensionais? Qual a diferença real entre eles?
- O que acontece se você alterar um valor no array retornado por ravel()?

11. Transposição e empilhamento

- Qual é o formato do array após a transposição?
- hstack aumenta o número de colunas ou de linhas?
- Quais as condições para que vstack e hstack funcionem corretamente?