

Lista 05 - Prog 01 2024

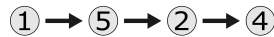
Emilio Vital Brazil

Entrega: 17 de junho 2024

Todas as questões abaixo serão consideradas usando a sintaxe **Python 3.10**.

1. Implemente uma lista encadeada em Python que tenha um método *inverte_ordem*. Este método deve receber uma lista encadeada como entrada e criar uma nova lista encadeada na qual todos os nós da lista original apareçam em ordem inversa (ver figura 1). Qual é a complexidade do método *inverte_ordem*?

Entrada:



Saída:



Figura 1: Exemplo de entrada e saída da função *inverte_ordem*.

2. Escreva um código que receba um inteiro no intervalo $[1, 3999]$ e converta ele em algarismos romanos.
3. Implemente a função *potencia*(*x*: float, *n*: inteiro), que calcula *x* elevado à potência *n* (ou seja, x^n).
 - (a) Faça uma solução em $\mathbf{O}(n)$
 - (b) Desafio - Faça uma solução em $\mathbf{O}(\log(n))$
4. Dado uma lista não vazia de números inteiros, onde cada elemento aparece duas vezes, exceto um. Encontre esse elemento único.

Você deve implementar uma solução com complexidade de tempo linear.
5. Escreva uma função *longest_common_prefix* que deve retornar o maior prefixo comum entre um lista de strings. Se não houver um prefixo comum, a função deve retornar uma string vazia . Assuma que todas as strings conterão apenas letras em caixa baixa. Além disso, analise a complexidade de pior caso da função.

Por exemplo:

```
>>> longest_common_prefix(['flower', 'flow', 'flight'])
'fl'
```