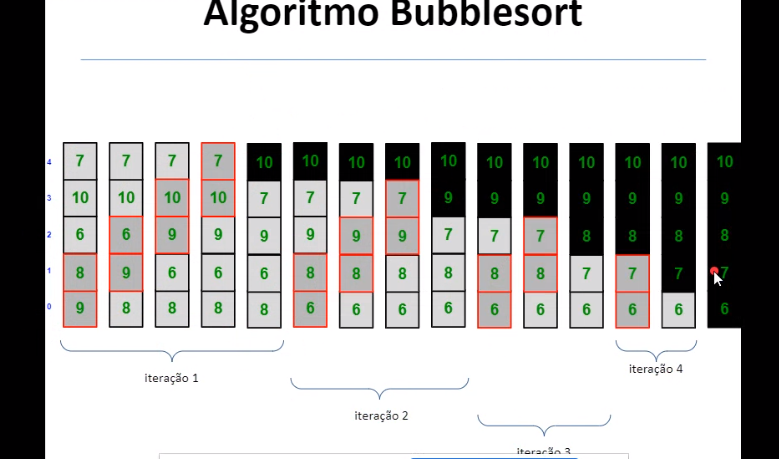
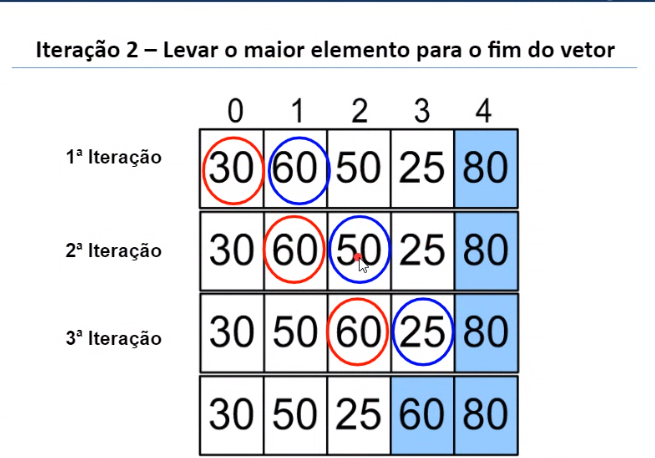
COMO FUNCIONA O ALGORITMO BUBBLESORT O(N²)?

Basicamente ele checa sempre de dois em dois e troca o maio com o menor, trocando suas posições caso isso ocorra.





No melhor caso, o bubblesort não troca nenhum número, mas faz as verificações. Podemos dizer que no melhor caso o custo é linear, ou seja O(N²)

No pior caso, o bubblesort troca todos, ou seja, é como se a lista tivesse inversa. Os items maiores estariam nas primeiras posições.

BubbleSort na prática:

def bubblesort(*vetor*):

    for i in range(0, len(*vetor*)-1): # Laço de "passadas"|

        for j in range(0, len(*vetor*)- 1 - i) : #  Exclui sempre o ultimo indice, que já é o maior da lista

            if *vetor*[j] > *vetor*[j + 1]:  #Verifica se a posição da esquerda é maior que a direita

                aux = *vetor*[j]  #Guardo a pos. da esquerda na minha variavel auxiliar

*vetor*[j] = *vetor*[j + 1] #Na minha posição da esquerdo atribuo o valor da direita

*vetor*[j + 1] = aux #Na posição da direita atribuo o antigo valor da pos. da esquerda

    return *vetor*

vetor = [1, 5, 9, 3, 10, 2, 15]

print(bubblesort(vetor))

Resumindo, o que esse algoritmo faz?

Este algoritmo consiste em dois laços e uma verificação. É preciso ter em mente que o laço de fora é responsável por contar as “passadas” ou iterações. Ele será o laço que não permitirá checarmos novamente o último índice da lista (Pois já sabemos que ele será o maior).