



{ Algoritmos E ESTRUTURA DE DADOS }



Busca Por Interpolação



```
while inicio <= fim and lista_codigos[inicio] <= chave <= lista_codigos[fim]:
    try:
        # Calcula a posição utilizando uma média ponderada das posições dos cod
        posicao = inicio + int(((chave - lista_codigos[inicio]) / (
            lista_codigos[fim] - lista_codigos[inicio])) * (fim - inicio))
    except ZeroDivisionError:
        return -1
    if lista_codigos[posicao] == chave:
        if contagem:
            end_time = time.time() - start_time
            print(f'Tempo de execução: {end_time} segundos')
        return posicao # Chave encontrada
    elif lista_codigos[posicao] < chave:
        inicio = posicao + 1
    else:
        fim = posicao - 1

return -1
```

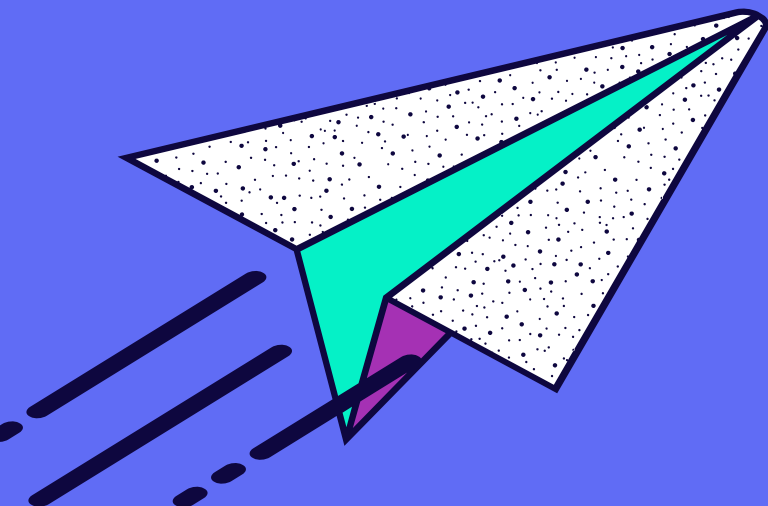
Conhecendo o algoritmo...

O algoritmo de busca por interpolação é uma técnica de busca em estruturas de dados ordenadas que usa uma fórmula de interpolação para estimar a posição do elemento desejado de forma mais precisa do que a busca binária tradicional.



Como funciona?

O algoritmo de busca por interpolação estima a posição do elemento desejado usando uma fórmula e ajusta essa estimativa com base nos valores dos elementos até encontrar o elemento desejado ou determinar sua ausência.



exemplo:

$$m = x - v[c] / v[f] - v[c] * (f - c) + c$$

$$m = 2 - 1 / 10 - 1 * (9 - 0) + 0$$

$$m = 1$$

0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9	10

0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9	10

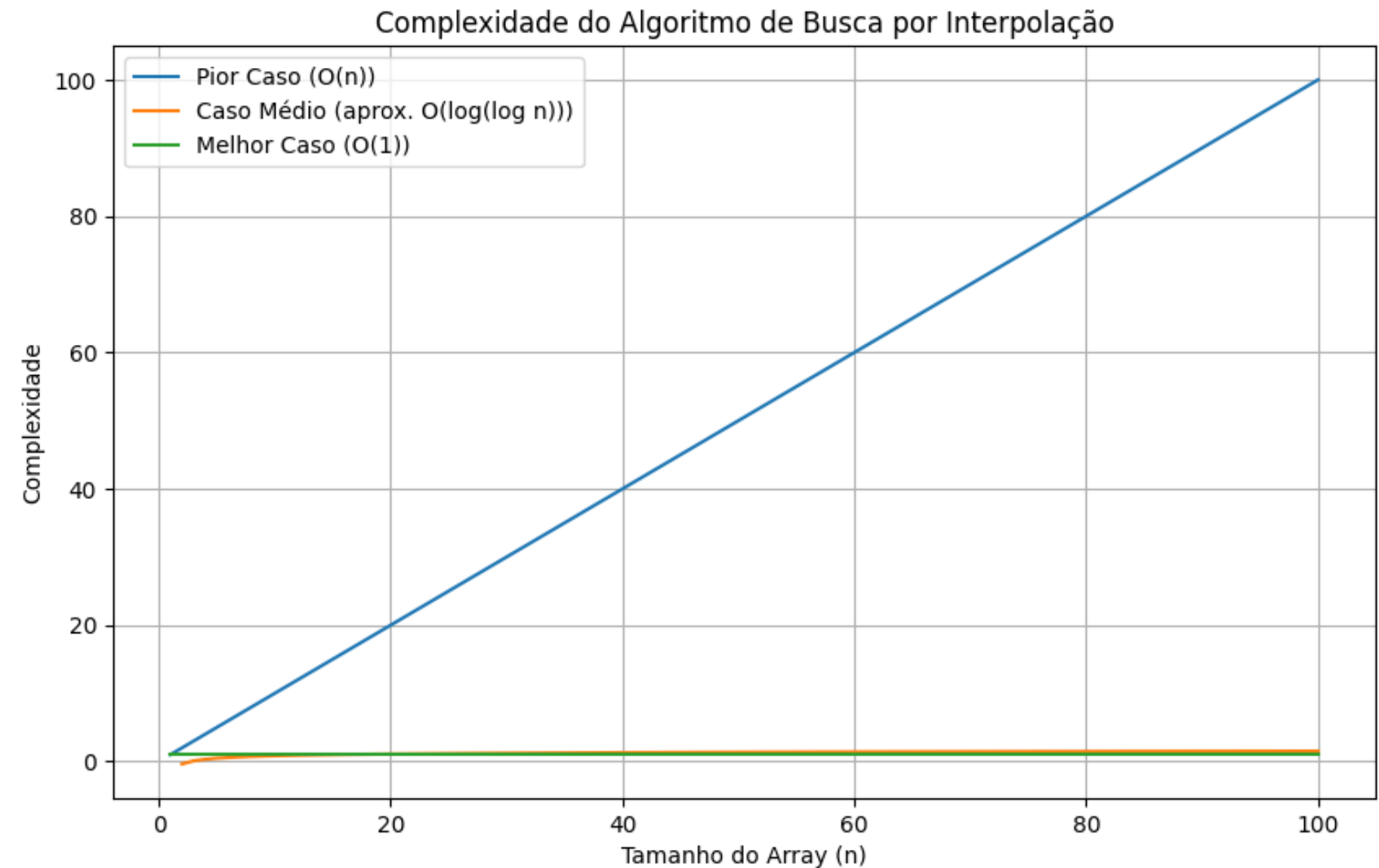
Porque e onde usar...

em geral a busca por interpolação tem as mesmas aplicações que a binária, no entanto conseguir ser mais rápida devemos levar em consideração que a utilização desse algoritmo só faz sentido se formos busca informações ordenadas, como dicionários, agendas e listas.



Complexidade

- O melhor caso do algoritmo de Busca por Interpolação ocorre quando o elemento procurado é o primeiro elemento considerado. complexidade $O(1)$
- O caso médio ocorrer dependendo do uniformidade dos dados se assumirmos que os dados estão uniformemente distribuídos, a complexidade média é aproximadamente $O(\log(\log n))$ ou melhor.
- O pior caso do algoritmo de Busca por Interpolação ocorre quando o arranjo de busca não possui uma distribuição uniforme dos elementos. Nesse cenário, o algoritmo tem um custo linear de comparações. $O(n)$



Vantagens

- **Eficiência em dados uniformemente distribuídos**
- **Desempenho em grandes conjuntos de dados.**

Desvantagens

- **Desempenho irregular com distribuição não uniforme**
- **Complexidade de implementação**



DEV

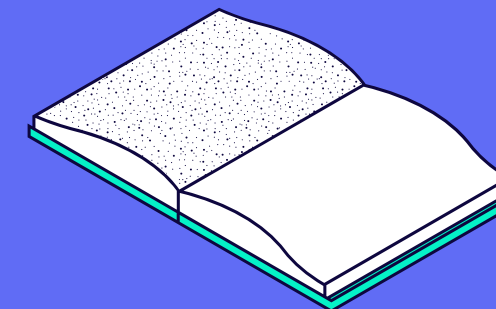
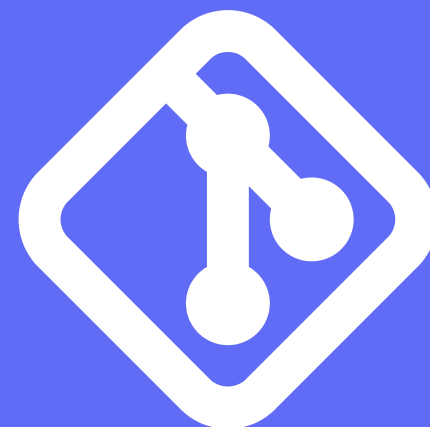


**como diria minha
ex, terminamos**



Reporsitório

<https://github.com/classroom-ufersa/buscaPorInterpolacaoClientes>



colaboradores

Antonio Vinícius Oliveira Lima

Eduardo Perreira Abrantes

João Gustavo Souza Lima

Marcelo Augusto Da Silva Fernandes

