

DEPOIS DA P1

> Lembrar de fragmentar os problemas para resolver parte por parte.

- Criar funções auxiliares
- Resolver o que eu conseguir pra conseguir pelo menos uma nota parcial.

30/06/2025

Verificar como que se fez a 4 da ultima lista
As condições de laço da professora geram:
enbile $i < \text{len}(\text{lst})$ and $\text{idx} \neq -1$

- Se o índice ultrapassar a lista
 - Se $i \neq -1$ (nº está na lista)
- } A repetição para.

Ordenar dados } Problemas básicos } Banco de
Procurar informação } da computação } dados.

- Se sabemos que o conjunto está ordenado, podemos otimizar a busca por meio de métodos de busca.

BUSCA SEQUENCIAL x BUSCA BINÁRIA

↳ n comparações

↳ $\log n$ comparações

sendo n o número de elementos no conjunto analisado.

def busca_binária (lst: list[int], n: int) -> int
Retorna o índice de um número buscado na lista. Se ele não
estiver na lista, é retornado -1.

i = 0

f = len(lst) - 1

idx = -1

while i < f and lst[m] != n

m = f // 2

elif lst[m] > n:
f = m - 1

elif lst[m] < n:
i = m + 1

else:

idx = m

return idx

'''

Exemplos:

>>> busca_binária.

Sempre que eu tiver que
usar um *lote* de
repetições dentro do outro,
é muito provável que
seja melhor *usar*
uma *função auxiliar*.

02/07/2025

Para os exercícios com repetição, na 5 tem que fazer uma auxiliar pra verificar se um n° tá na lista. Na 8 usa essa mesma função de verificar na lista pra colocar as do intervalo, com append no resultado, que não estão na lista dada como parâmetro.

Na 9 dá pra gerar uma lista com o comprimento das sequências de 1s. Depois roda um loop de repeat pra averiguar qual deles é o maior.