



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO

Trabalho de Matemática Discreta.

Professor Jefferson Moraes

Busca binária usando método iterativo e recursivo.

Raiane Gabriele Brito de Oliveira.

## 1 INTRODUÇÃO

A busca binária é um algoritmo fundamental em ciência da computação, utilizado para encontrar a posição de um elemento em um conjunto ordenado de dados. A análise de desempenho dessas implementações é crucial para compreender como o tempo de execução varia em relação ao tamanho do conjunto de dados. O trabalho consiste em desenvolver a questão 14 da lista 5 de matemática discreta:

“14. Escreva um algoritmo (em pseudocódigo) que realize busca binária de forma iterativa e o implemente numa linguagem de programação a sua escolha. Construa um gráfico mostrando a relação do valor de entrada x tempo de execução do algoritmo. Considerando uma análise assintótica em pior caso, explique se o desempenho do algoritmo é superior, inferior ou igual ao do algoritmo que implementa busca binária de forma recursiva. “

## 2 DESENVOLVIMENTO

Pseudocódigo para a busca binária iterativa:

```
10 Função buscaBinariaIterativa(arr, x):
11     início = 0
12     fim = tamanho(arr) - 1
13
14     enquanto início <= fim:
15         meio = (início + fim) / 2
16
17         se arr[meio] é igual a x:
18             retorne meio
19         senão, se arr[meio] < x:
20             início = meio + 1
21         senão:
22             fim = meio - 1
23
24     retorne -1
```

Pseudocódigo para a busca binária recursiva:

```
10 Função busca_binaria_recursiva(array, valor, inicio, fim):
11   Se inicio > fim:
12     Retorne -1 # Elemento não encontrado
13   Senão:
14     Meio = (inicio + fim) / 2
15     Se array[meio] == valor:
16       Retorne meio # Elemento encontrado
17     Se array[meio] > valor:
18       Retorne busca_binaria_recursiva(array, valor, inicio, meio - 1)
19     Senão:
20       Retorne busca_binaria_recursiva(array, valor, meio + 1, fim) #
```

### 3. IMPLEMENTANDO EM PYTHON:

Repositório do GitHub: <https://github.com/Flaeury/busca-binaria>

Neste código, foi necessário testar 3 vezes vetores de 10 a 50 milhões e tirar a média do tempo de execução. Após isso, o arquivo `main.py` guarda numa lista o tamanho dos vetores e o resultado da média de ambos os casos. Chamamos a biblioteca `matplotlib` para visualizar o gráfico.

### 4. ANÁLISE DE DESEMPENHO:

A busca binária, tanto iterativa quanto recursiva, possui uma complexidade de tempo de  $O(\log n)$  no pior caso, onde  $n$  é o tamanho do array. A busca binária é mais eficiente do que uma busca linear ( $O(n)$ ) para arrays grandes. Ambos os algoritmos, iterativo e recursivo, têm a mesma complexidade assintótica. A escolha entre eles muitas vezes depende das preferências do programador ou da situação específica do problema. A implementação iterativa pode ser mais eficiente em termos de uso de memória, pois evita o acúmulo de chamadas recursivas. Contudo, a busca binária recursiva é frequentemente considerada mais elegante e fácil de entender.