**Atividade B3-2 – Comparando Eficiência de Algoritmos**

Analisando a busca linear, a busca linear em ordem e a busca binária. A análise será realizada tendo em conta o número de etapas necessárias para cada um dos casos:

1. Busca Linear (A, n, x): Percorre todos os elementos da lista A até encontrar o elemento “x” ou até percorrer toda a lista.
2. Busca Linear em Ordem (A, n, x): Percorre a lista de forma linear, mas interrompe a busca assim que encontra um elemento maior que “x”, assumindo que a lista está ordenada.
3. Busca Binária (A, n, x): Divide a lista pela metade a cada iteração, buscando “x” em uma lista ordenada.

Vamos analisar cada cenário (coluna da tabela) para cada algoritmo:

**X**  **A** - BuscLinear:

A busca linear vai iterar até encontrar x e, em média, espera-se que o elemento seja encontrado na metade da lista.

* Busca Linear em Ordem:

Mesmo que a lista esteja ordenada, a busca linear em ordem tem o mesmo comportamento que a busca linear.

* Busca Binária:

A busca binária divide o problema pela metade a cada iteração. O tempo de execução é log2(n), pois é o número de vezes que podemos dividir n por 2 até chegar a 1.

**X = A[1]** –

**- Busca linear**:

Apenas 1 passo é necessário.

* Busca Linear em Ordem:

Igual na busca linear, apenas 1 passo é necessário.

* Busca Binária:

A busca binária, ainda precisa calcular o índice médio e fazer comparações, resultando em um tempo de execução constante.

**X = A[n]**

- Busca linear

A busca linear precisa percorrer todos os elementos.

* Busca Linear em Ordem:

Mesmo que a lista esteja ordenada, a busca linear em ordem ainda precisa percorrer todos os elementos.

* Busca Binária:

A busca binária é mais eficiente, mas ainda levará log2(n) passos.

**X** ∉ **A**

- Busca linear

A busca linear precisa percorrer todos os elementos e mesmo assim, não encontra o x.

* Busca Linear em Ordem:

A busca linear em ordem pode sair antes, se encontrar um elemento maior que x. No pior caso, precisará percorrer **n** elementos.

* Busca Binária:

A busca binária precisará dividir a lista até que o intervalo se esgote, resultando em log2(n) passos.

|  |  |  |  |
| --- | --- | --- | --- |
| Situação | Busca Linear | Busca Linear em Ordem | Busca Binária |
| X ∈ A | 2t + 5px - 2 | 2t + 5tn+(2tn - 1) | Log2(n) |
| X = A[1] | 5t | 7t | Log2(n) |
| X = A[n] | 2t + 3tn+ (2tn-1) | 2t + 5tn + (2tn - 1) | Log2(n) |
| X ∉ A | 2t + (3tn+1t) + (2tn-1t) | 2t + 5tn + (2tn – 1) | Log2(n) |