Researching Algorithms

With Samples

**Linear Search (sequential search)**

With a linear search, we iterate through a collection one element at a time.

Runtime complexity: O(n)



Disadvantages:

* Slow for large datasets

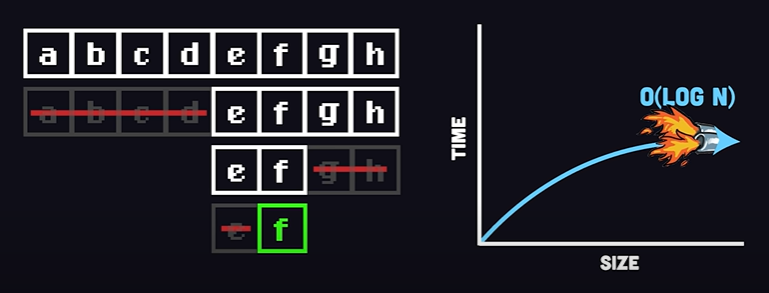
Advantages:

* Fast for searches of small to medium datasets
* Does not need to sorted (adv. over binary and interpolation searches)
* Useful for data structures that do not have random access (Linked list)

Imagine we are looking for a word in dictionary and we start looking from page 1 and continue till finding.

**Binary Search (sequential search)**

Imagine we are looking for a word in a dictionary, but this time we divide dictionary into halves to find it fast.



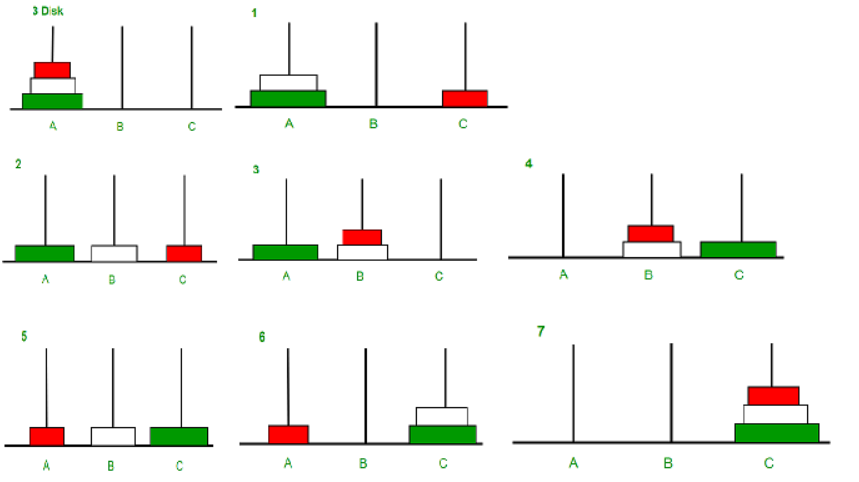
**Hanoi Towers Algorithm**



The Tower of Hanoi is a mathematical game or puzzle consisting of three rods and a number of disks of various diameters, which can slide onto any rod. The puzzle begins with the disks stacked on one rod in order of decreasing size, the smallest at the top, thus approximating a conical shape. The objective of the puzzle is to move the entire stack to one of the other rods, obeying the following rules:

* Only one disk may be moved at a time.
* Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack or on an empty rod.
* No disk may be placed on top of a disk that is smaller than it.

With three disks, the puzzle can be solved in seven moves. The minimal number of moves required to solve a Tower of Hanoi puzzle is 2n − 1, where n is the number of disks. (<https://www.youtube.com/watch?v=YstLjLCGmgg&t=190s> )



**Ternary Search Algorithm**

In Ternary search, the array must be sorted. Let’s imagine we are given the list below:

List = [10, 20, 30, 40, 50, 60, 70, 80]

We have to find number 10

In this search algorithm we need 2 **mids**.

First mid is,

]

*l = 0; r = 7*

Second mid is

]

*r =7; l = 0*

Since we are trying to find 10, and 10 is smaller than 30 (m1), it must be in first group. So we move r to left. And we repeat this process.

Let’s try to find number 45. M1 is not 45, m2 is not 45. So 45 must be in second group. (40 is l, 50 is r here). But 45 is between 40 and 50. So we move r to left and l to right. Here 45 does no exits in our list, so it must return False.

**SOURCES**

1. Linear Search - <https://geeksforgeeks.org/linear-search/>
2. Linear Search - <https://www.youtube.com/watch?v=246V51AWwZM>
3. Binary Search - <https://www.geeksforgeeks.org/binary-search/>
4. Binary Search - <https://www.youtube.com/watch?v=MFhxShGxHWc>
5. Hanoi Towers - <https://www.geeksforgeeks.org/c-program-for-tower-of-hanoi/>
6. Hanoi Towers - <https://www.youtube.com/watch?v=YstLjLCGmgg>
7. Ternary Search - <https://www.youtube.com/watch?v=o3HPRpbGlbI>
8. Ternary Search - <https://www.geeksforgeeks.org/ternary-search/>
9. Fibonacci Algorithm - <https://www.geeksforgeeks.org/fibonacci-search/>
10. Linear and Binary difference - <https://www.youtube.com/watch?v=sSYQ1H9-Vks>
11. Binary and Ternary difference - <https://www.youtube.com/watch?v=alpGaebSZFk&t=145s>