Case Analysis

Mohammed Rizin
Unemployed

March 21, 2025

1 Linear Search Algorithm

Suppose we have an array:

8	6	12	5	9	7	4	3	16	18
0	1	2	3	4	5	6	7	8	9

$\underbrace{\quad}_{key}$

Figure 1: Searching for key = 3 after 7 steps

Algorithm 1 Linear Search Algorithm

```
1: procedure LinearSearch(A, n, x)
2:     for i ← 0 to n − 1 do
3:         if A[i] = x then
4:             Return i
5:         end if
6:     end for
7:     Return −1
8: end procedure
```

8	6	12	5	9	7	4	3	16	18
0	1	2	3	4	5	6	7	8	9

Not Found!

Figure 2: Searching for key = 0 after 10 steps

Best Case $B(n) \implies$ Searching key element is present at index 0. Then it will take $O(1)$.
Worst Case $W(n) \implies$ Either the element is absent or its at last index. Then it will take $O(n)$.
Average Case $\implies \frac{\text{all possible cases time}}{\text{no. of cases}}$

Most of the cases, finding the average case time is not possible. But most of the cases it would be equal to Worst case time.

To calculate :

$$AvgTime = \frac{\sum_{i=0}^{n} \text{time taken to find } i^{th} \text{ index element}}{n}$$

$$AvgTime = \frac{1 + 2 + 3 + \cdots + n}{n}$$

$$AvgTime = \frac{\frac{n(n+1)}{2}}{n}$$

$$AvgTime = \frac{(n+1)}{2} \simeq O(n) \cong W(n)$$