Algorithm Complexity & Time Space Trade Off ???

Algorithm: Finite sequence of steps/instructions used to solve a particular problem.

Algorithm Complexity :Complexity of an algorithm is a function f(n) which measures the amount of running time or space needed to execute an algorithm in terms of Input size.

It is the function which gives the running time or space in terms of input size.

- Space Complexity
- Time Complexity

Algorithm Complexity & Time Space Trade Off ???

- **Space Complexity** :It is the minimum memory required by an algorithm to produce output .

The Space is measured by counting the total memory needed by algorithm in terms of variables used by algorithm.

Algorithm Complexity & Time Space Trade Off ???

Time Complexity :it is the amount of computational time it needs to completion.

It can be expressed as a function of no. of key operations performed.

Time is directly proportional to number of key operations.

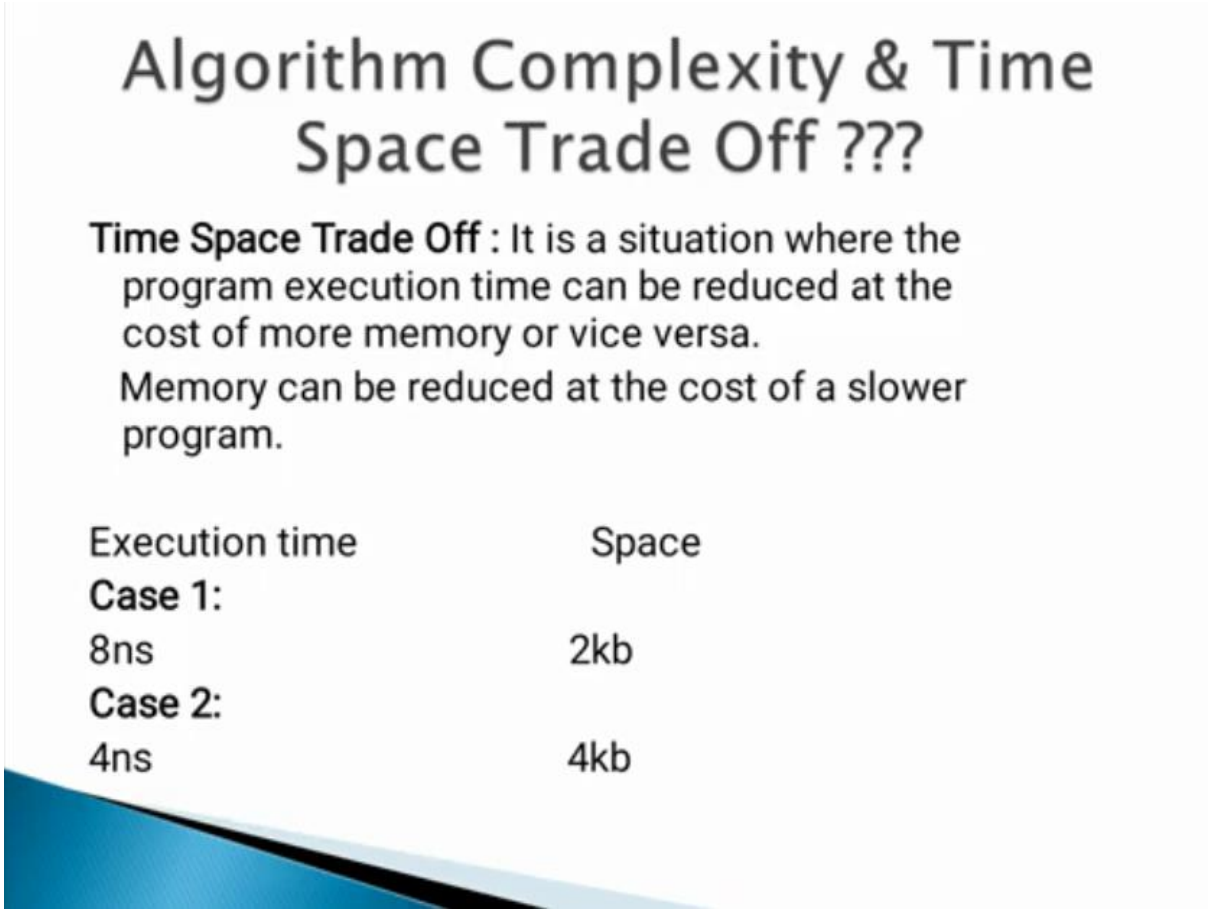
$T \propto$ No. of key operations

Algorithm Complexity & Time Space Trade Off ???

1
Log n
n
n log n
n^2
n^3
n!

High (fast)

Low (Slow)



Searching:

- Linear O(n) Works on both sorted and unsorted data
- Binary O(log n) Works on sorted data only

Sorting:

- Merge O(n log n) Space Complexity: O(n)
- Heap O(n log n) Space Complexity: O(1)
- Quick O(n log n), O(n log n), $O(n^2)$ Space Complexity: O(n log n)

Example:

- Data: ID, Name, Location
- Search by ID/Name
- Sort by ID/Name
- Solution?