GCIS.123.601

QUICK_SORT

DONE BY:SAMIRA ALSAQQA,MOHAMMED ABUJBARA,MARYAM SABT AND SWATI POJARY

GROUP#10

STEP 01

TABLE OF CONTENTS

STEP 02

GITHUB REPOSITORIES

- Mohammed Abujbara: https://github.com/MohamadAbujbara/mhmd-s-repo
- Samira AlSaqqa: https://github.com/SamiraAlsaqqa/gcis123
- Swati Pojaryhttps: //github.com/sw4tii/gcis123.git
- Maryam Sabt: https://github.com/maryamsabt/Maryam.git

STEP 03

WHATS QUICK SORT

A FAST AND EFFICIENT SORTING ALGORITHM

QUICK SORT

Quick Sort is renowned for its efficiency and speed, making it one of the most efficient sorting algorithms available.

IMPORTANCE OF SORTING

- Sorting plays a crucial role in various applications, such as searching, data analysis, and organizing databases.

STEP 04

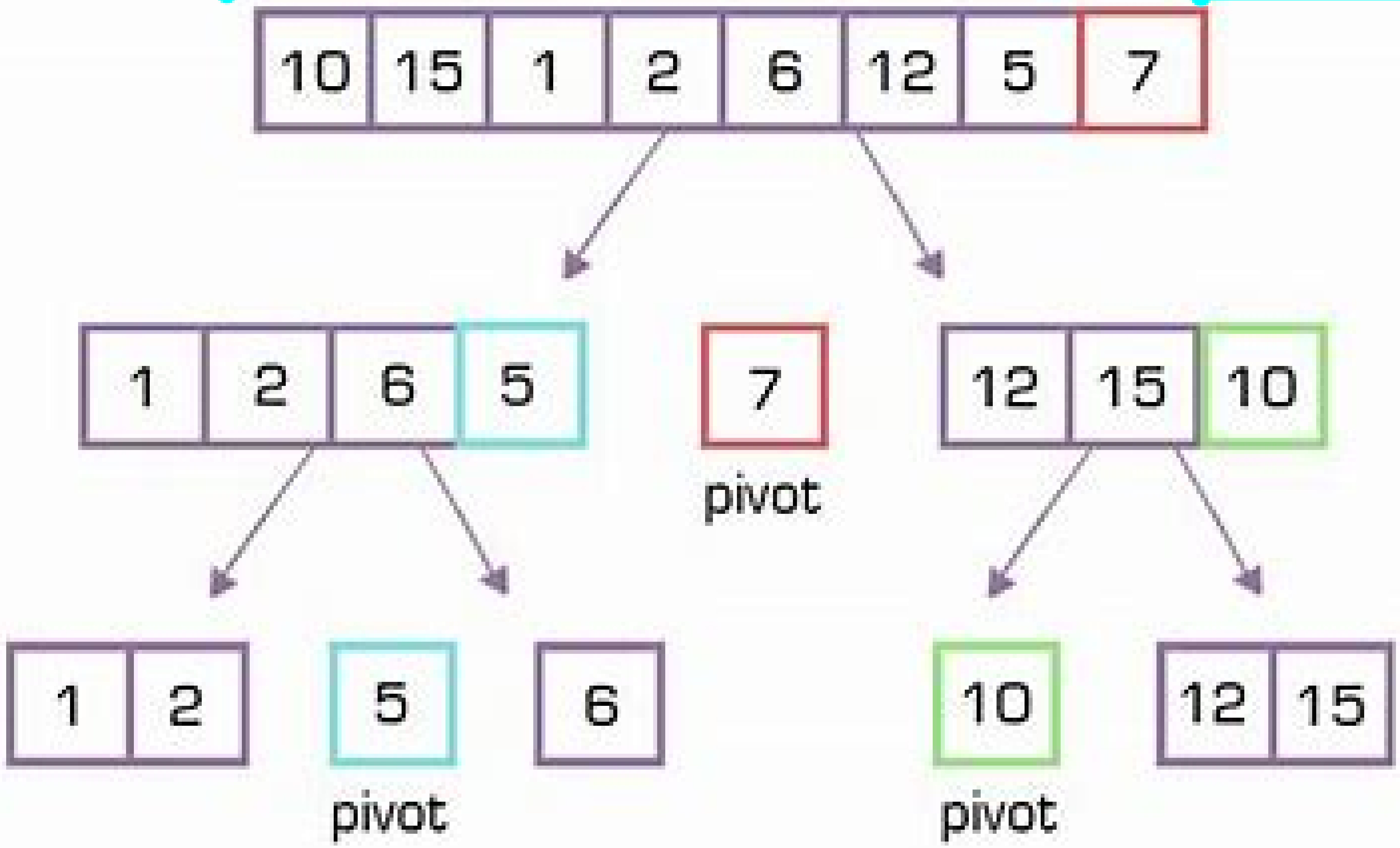
QUICKSORT ALGORITHM

```
           Quick sort
------------------------------
How many numbers you war

Enter 7 elements:
 5
 9
 23
 11
 27
 6
 16

Sorted elements after ap

5 6 9 11 16 23 27
```

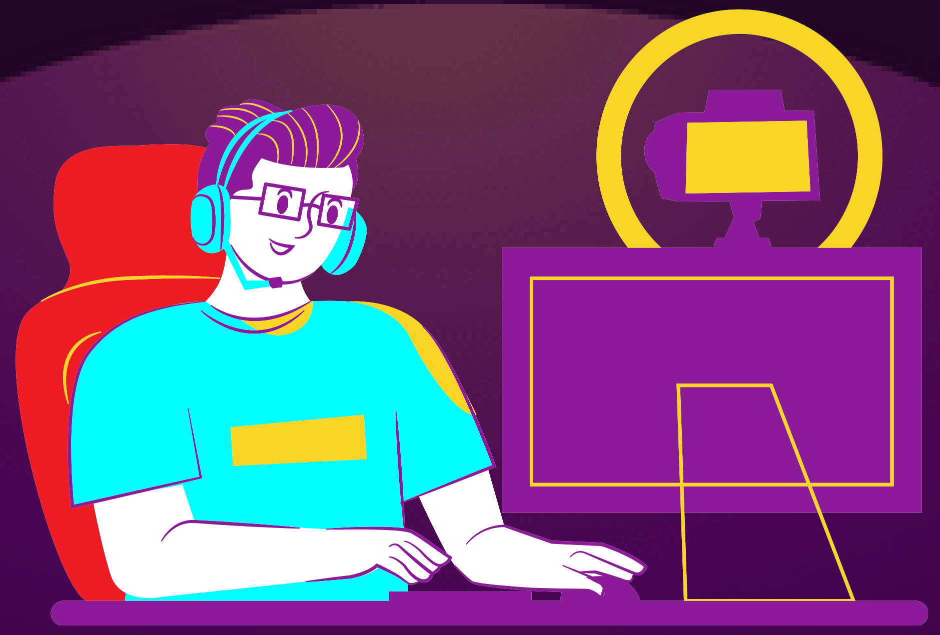
QUICK SORT IS A DIVIDE AND CONQUER ALGORITHM WHICH RELIES ON A PARTITION OPERATION:

TO PARTITION AN ARRAY AN ELEMENT CALLED A PIVOT IS SELECTED

- All elements smaller than the pivot are moved before it and all greater elements are moved after it
- The lesser and greater sublists are then recursively sorted

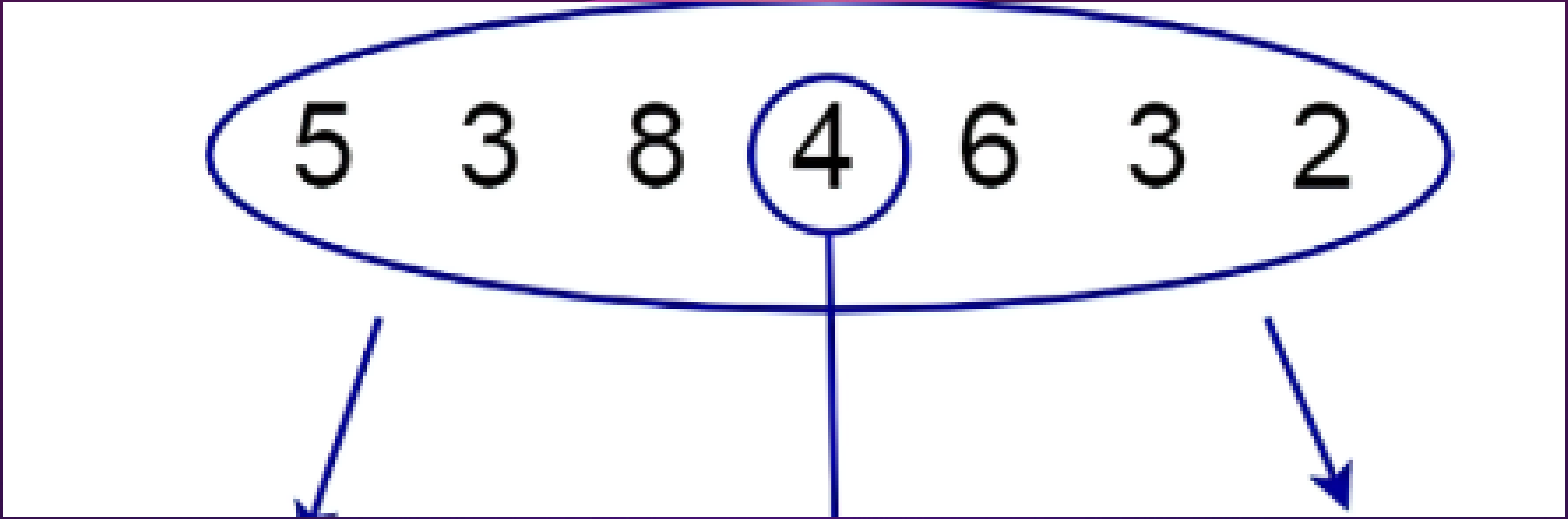

10
15
1
2
6
12
5
7
1
2
6
5
7
pivot
12
15
10
1
2
5
pivot
6
10
pivot
12
15

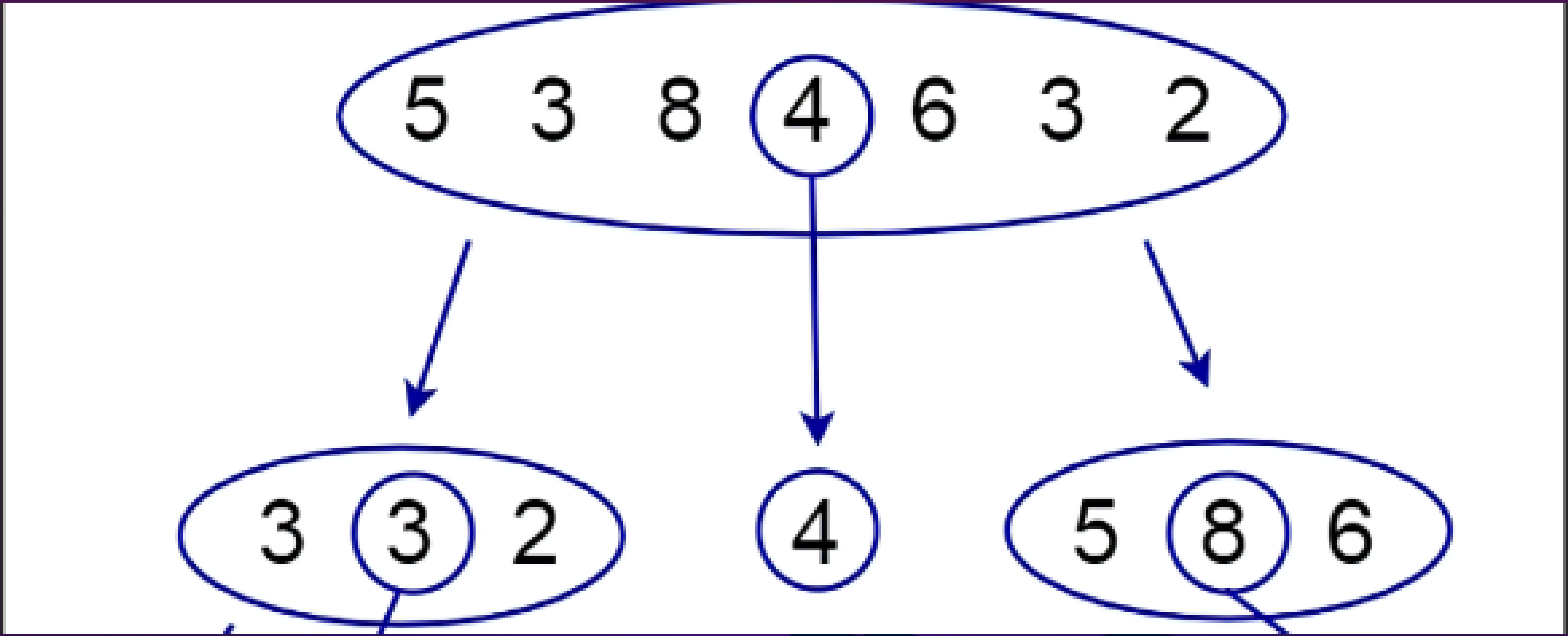
STEP 05

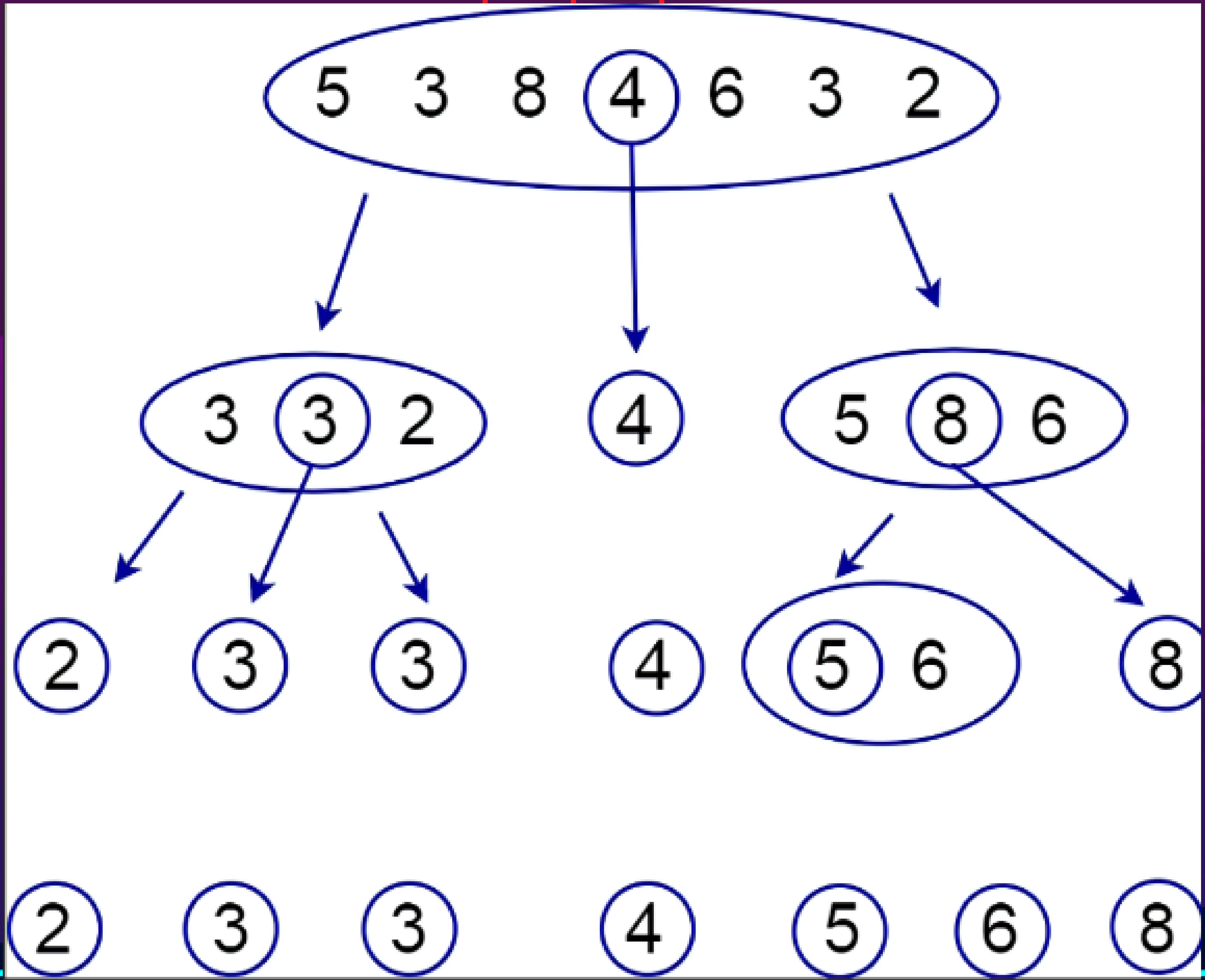


PIVOT

1. Correct position in final, sorted array
2. Items to the left are smaller
3. Items to the right are larger


5 3 8 4 6 3 2


5 3 8 4 6 3 2
3 3 2
4
5 8 6


5 3 8 4 6 3 2
3 3 2
4
5 8 6
2 3 3 4 5 6 8
2 3 3 4 5 6 8

STEP 06

DIFFERENCE BETWEEN QUICK SORT & MERGE SORT

QUICK SORT

Approach: Divide-and-conquer.

Pivot Selection: Crucial for performance.

Space Complexity: Typically O(log n), but can degrade to O(n).

In-Place: Yes.

Stability: Not stable.

MERGE SORT

Approach: Divide-and-conquer.

Merge Step: Key operation.

Space Complexity: Always O(n).

In-Place: No.

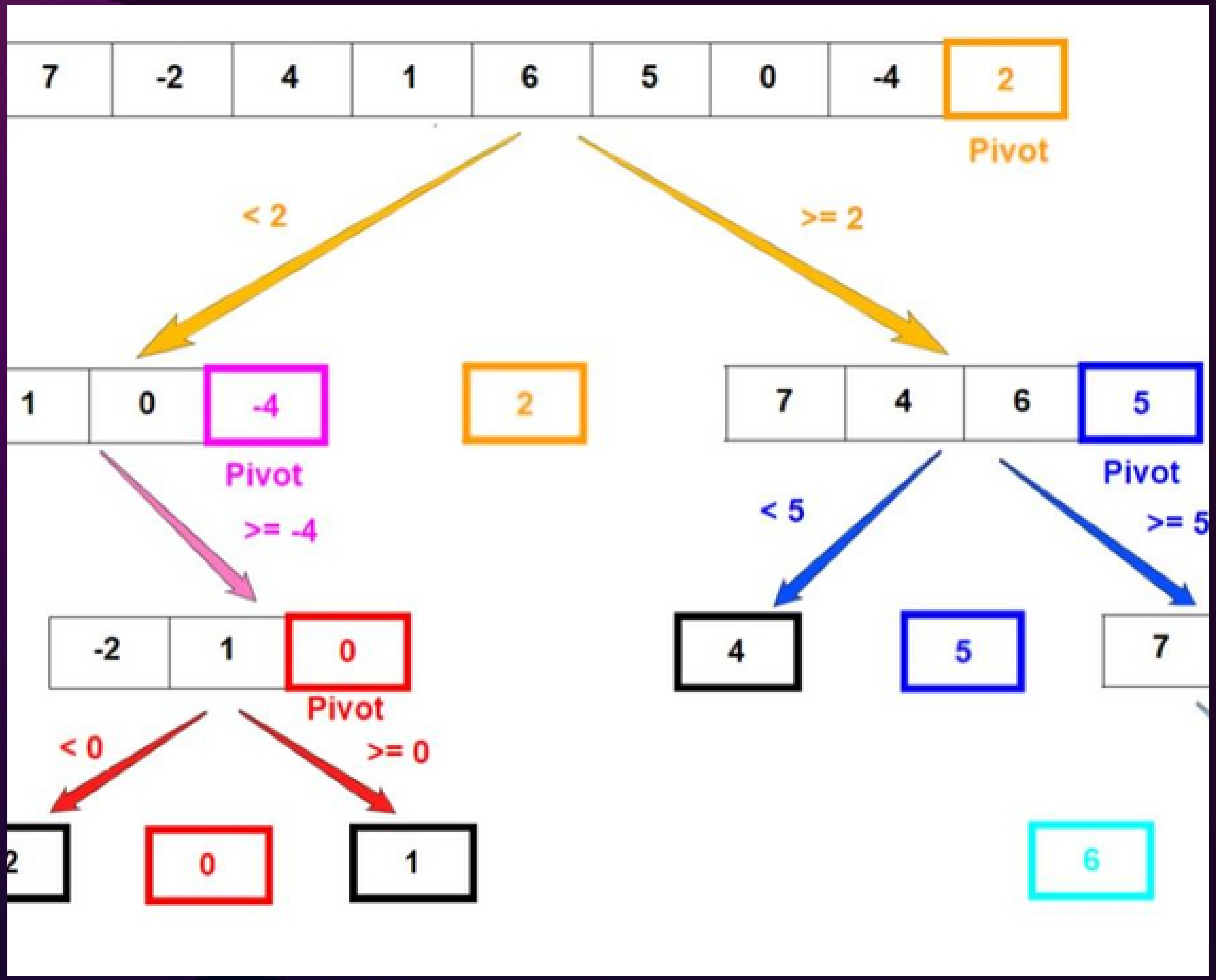
Stability: Stable.

COMPARISON

- Quick sort's efficiency depends on pivot selection, while merge sort consistently performs in O(n log n).
- Quick sort is typically more space-efficient.
- Merge sort is stable, while quick sort is not inherently stable.

ADVANTAGES OF QUICK SORT

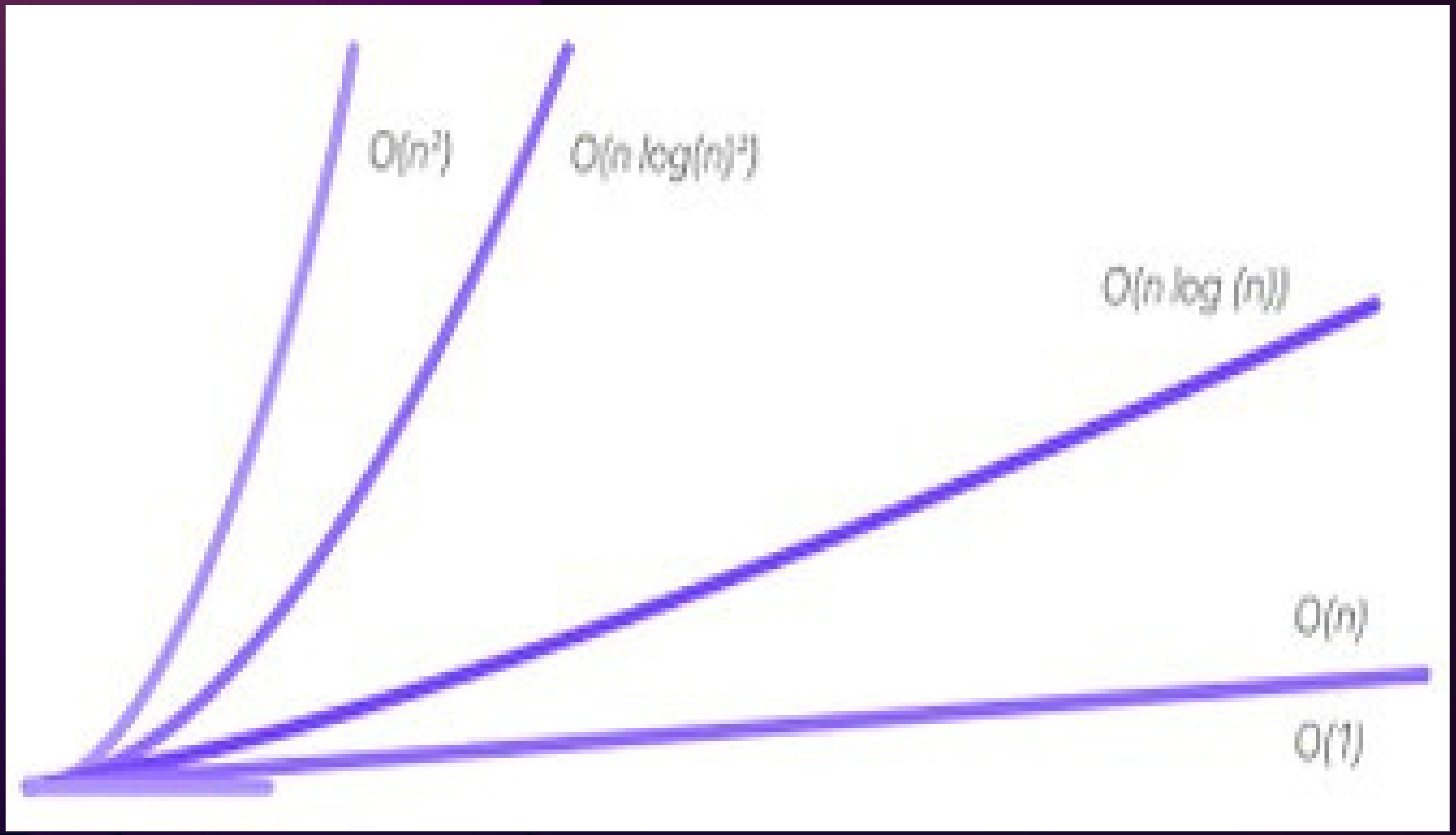
- Speed: Quick Sort is exceptionally fast and efficient, especially for large datasets.
- In-Place Sorting: It's an in-place sorting algorithm, which means it doesn't require additional memory.
- Practical Usage: Quick Sort is widely used in real-world applications due to its favorable average-case performance.



BEST CASE SCENARIO

The best-case scenario time complexity occurs when the pivot choice equally divides the arrays into equal sized sub-arrays during partitioning.

Time Complexity: O(nlog(n))



STEP 09

APPLICATIONS OF QUICKSORT

Some real-world applications of QuickSort include:

- In computer graphics, QuickSort is used for image rendering.
- In addition, it is used for data visualization.
- In numerical computations, QuickSort is used for matrix sorting

THANK YOU

I HOPE YOU LEARNED SOMETHING NEW!