

## Computer Science and Engineering > [Data Structures – 1](#) > [Experiments](#)

[Aim](#)[Overview](#)[Recap](#)[Pretest](#)[Linear Search](#) ▾[Binary Search](#) ▾[Aim](#)[Concept](#)[Algorithm](#)[Demo](#)[Practice](#)[Exercise](#)[Quiz](#)[Analysis](#) ▾[Posttest](#)[Further Readings/References](#)[Feedback](#)

### Unsorted Arrays vs Binary Search

Choose

difficulty:



Beginner



Intermediate



Advanced

1. What is the prerequisite to perform Binary Search?

☐ a: The values in the array have a maximum bound value they can take

[Explanation](#)

☒ b: Array must be sorted either in ascending or descending order

[Explanation](#)

☐ c: Array must be broken into sub-arrays

☐ d: None of the above

2. Binary Search is an example of \_\_\_\_\_ algorithm.

☐ a: Greedy

☐ b: Dynamic Programming

☐ c: Backtracking

☒ d: None of the above [Explanation](#)

3. Let us assume an array [1,23,145,178,1203]. How many iterations are needed to find 23? [Assuming we are considering floor of values for floating point values, and index starting from 1]

☒ a: 3 [Explanation](#)

☐ b: 4

☐ c: 2

☐ d: 5

4. Let us assume an array [11, 33, 145, 1294, 1356, 1450, 3300, 4500, 6000, 8000, 9000]. Let us search for 4500 using binary search. What would be the mid values at the second and third iteration respectively? [Assuming we are considering floor of values for floating point values, and index starting from 1]

☐ a: 1450 and 6000

☒ b: 6000 and 3300 [Explanation](#)

☐ c: 6000 and 4500

☐ d: 1450 and 4500

5. What is the space complexity of binary search implemented using recursion?

☐ a:  $O(1)$

☐ b:  $O(N)$

☐ c:  $O(N \log N)$

☒ d:  $O(\log N)$  [Explanation](#)

