3. Why is the complexity of binary search O(logN) where N is the size of the list? Explain.

**Answer:**

We know that Binary Search can be applied only on sorted data structure (we will use Array for example).

In binary search for each iteration we find the mid of the array then we check whether our desired value is in the mid, if not we go either to the right or left of the mid depending on whether our desire value is larger or smaller than mid. We do the same thing until we find the desired value or our logic for absence of desired value is met.

In each iteration our searching area becomes half. For this reason despite the size being N we don’t have to check the whole array for our desired value. In each iteration we shorten our scope to half which leads to the time complexity of O(logN)