Statement:

1. **Define a Function:**

Create a function findRotationIndex(arr, left, right) to recursively search for the rotation index.

Base Case:

- If the subarray has only one element, return its index since it is the minimum.

Check if Subarray is Already Sorted:

- If arr[left] <= arr[right]
- it means the array segment is already sorted. In this case, arr[left] is the smallest element, and its index is the rotation index.

Calculate the Middle Index:

- Compute mid = left + (right - left) / 2.

Compare Middle with its Neighbors:

- If arr[mid] > arr[mid + 1], then mid + 1 is the rotation index (smallest element).
- If arr[mid] < arr[mid - 1], then mid is the rotation index (smallest element).

Decide Which Side to Search:

- If arr[mid] >= arr[left], the left part is sorted, so search the right half.
- Otherwise, search the left half.