# Test data

## Test Case 1: kth Smallest in S1

**Array:** [7, 4, 3, 6, 1]  
**k:** 3  
**Expected Output:** 4

### Explanation:

* The array is partitioned using 7 as the pivot.
* S1 (elements less than or equal to the pivot) will be [4, 3, 6, 1].
* there is no S2 as all elements are lesser than pivot
* the expected output is therefore in S1, which is 4

## Test Case 2: kth Smallest is the Pivot

**Array:** [6, 3, 1, 10, 8, 4]  
**k:** 4  
**Expected Output:** 6

### Explanation:

* The array is partitioned using 6 as the pivot.
* S1 (elements less than or equal to the pivot) will be [3, 1, 4].
* The pivot is 6, which is the 4th smallest element.

## Test Case 3: kth Smallest in S2

**Array:** [1, 3, 5, 2, 19, 13, 4]  
**k:** 6  
**Expected Output:** 13

### Explanation:

* The array is partitioned using 1 as the pivot initially, and the process continues recursively.
* S1 (elements less than or equal to the pivot) will be [1].
* Since k > size of S1 + 1, we look in S2.
* After several partition steps, the 6th smallest element is found to be 13.