**[Kth Largest Element in an Array](https://leetcode.com/problems/kth-largest-element-in-an-array/)**

**package** leetcodeMedium;

**public** **class** KthLargestElement {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**int**[] nums = {3,2,3,1,2,4,5,5,6};

System.***out***.println(*findKthLargest*(nums, 4));

}

**public** **static** **int** findKthLargest(**int**[] nums, **int** k) {

**int** start = 0;

**int** end = nums.length - 1;

**int** K = nums.length - k;

**while**(start < end) {

**int** pivot = *partition*(nums , start , end);

**if**(pivot == K) {

**return** nums[pivot];

}

**else** **if**(pivot > K) {

end = pivot - 1;

}

**else** {

start = pivot + 1;

}

}

**return** nums[start];

}

**public** **static** **int** partition(**int**[] nums , **int** i , **int** j) {

**int** pivot = i;

**while**(i <= j) {

**while**(i <= j && nums[pivot] >= nums[i]) {

i++;

}

**while**(i <= j && nums[pivot] <= nums[j]) {

j--;

}

**if**(i < j) {

*swap*(nums , i , j);

}

}

*swap*(nums , pivot , j);

**return** j;

}

**public** **static** **void** swap(**int**[] nums , **int** i , **int** j) {

**int** temp = nums[i];

nums[i] = nums[j];

nums[j] = temp;

}

}

Time complexity : Average(n), n is number of elements in array

Space Complexity : O(1) constant space