Problem 1984: Minimum Difference Between Highest and Lowest of K Scores

Problem Information

Difficulty: Easy
Acceptance Rate: 59.29%
Paid Only: No
Tags: Array, Sliding Window, Sorting

Problem Description

You are given a **0-indexed** integer array `nums`, where `nums[i]` represents the score of the `ith` student. You are also given an integer `k`.

Pick the scores of any `k` students from the array so that the **difference** between the **highest** and the **lowest** of the `k` scores is **minimized**.

Return _the**minimum** possible difference_.

Example 1:

Input: nums = [90], k = 1 **Output:** 0 **Explanation:** There is one way to pick score(s) of one student: - [**_90_**]. The difference between the highest and lowest score is 90 - 90 = 0. The minimum possible difference is 0.

Example 2:

Input: nums = [9,4,1,7], k = 2 **Output:** 2 **Explanation:** There are six ways to pick score(s) of two students: - [**_9_** ,**_4_** ,1,7]. The difference between the highest and lowest score is 9 - 4 = 5. - [**_9_** ,4,**_1_** ,7]. The difference between the highest and lowest score is 9 - 1 = 8. - [**_9_** ,4,1,**_7_**]. The difference between the highest and lowest score is 9 - 7 = 2. - [9,**_4_** ,**_1_** ,7]. The difference between the highest and lowest score is 4 - 1 = 3. - [9,**_4_** ,1,**_7_**]. The difference between the highest and lowest score is 7 - 4 = 3. - [9,4,**_1_** ,**_7_**]. The difference between the highest and lowest score is 7 - 1 = 6. The minimum possible difference is 2.

Constraints:

* `1 <= k <= nums.length <= 1000` * `0 <= nums[i] <= 105`

Code Snippets

C++:

```
class Solution {
public:
int minimumDifference(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public int minimumDifference(int[] nums, int k) {

}
}
```

Python3:

```
class Solution:
def minimumDifference(self, nums: List[int], k: int) -> int:
```