

## Smart Coding & Interview Series

### Top-20 Basic Program (Selection & Applications)

---

First, understand the solution building strategies and coding for the problems in LIVE/VIDEO session and then you apply those strategies discussed in LIVE/VIDEO session to solve the following problems. Use your favourite language(C/C++/Java/C#/Python/Scala) for coding.

**1) Top K Frequent Words:** Given a non-empty list of words, return the  $k$  most frequent elements. Your answer should be sorted by frequency from highest to lowest. If two words have the same frequency, then the word with the lower alphabetical order comes first.

**Example:**

Input: ["i", "love", "leetcode", "i", "love", "coding"],  $k = 2$

Output: ["i", "love"]

Explanation: "i" and "love" are the two most frequent words.

Note that "i" comes before "love" due to a lower alphabetical order.

**Source:** <https://leetcode.com/problems/top-k-frequent-words/>

**2) Top K Frequent Elements:** Given a non-empty array of integers, return the  $k$  most frequent elements.

**Example:**

Input:  $nums = [1,1,1,2,2,3]$ ,  $k = 2$

Output: [1,2]

**Source :** <https://leetcode.com/problems/top-k-frequent-elements/>

**3) K Closest Points to Origin:** We have a list of points on the plane. Find the  $K$  closest points to the origin (0, 0). (Here, the distance between two points on a plane is the Euclidean distance.) You may return the answer in any order. The answer is guaranteed to be unique (except for the order that it is in.)

**Example:**

Input:  $points = [[1,3],[-2,2]]$ ,  $K = 1$

Output:  $[[1,3]]$

**Source:** <https://leetcode.com/problems/k-closest-points-to-origin/>

**4) Kth Largest Element in an Array:** Find the  $k$ th largest element in an unsorted array. Note that it is the  $k$ th largest element in the sorted order, not the  $k$ th distinct element.

**Example:**

Input: [3,2,1,5,6,4] and  $k = 2$

Output: 5

**Source:** <https://leetcode.com/problems/kth-largest-element-in-an-array/>

## Smart Coding & Interview Series

### Top-20 Basic Program (Selection & Applications)

---

**5) Third Maximum Number:** Given a **non-empty** array of integers, return the third maximum number in this array. If it does not exist, return the maximum number. The time complexity must be in  $O(n)$ .

**Example:**

Input: [3, 2, 1]

Output: 1

**Source:** <https://leetcode.com/problems/third-maximum-number/>