

Kick Start 2017 - Round B

Math Encoder

Problem

Professor Math is working on a secret project and is facing a challenge where a list of numbers need to be encoded into a single number in the most efficient manner. After much research, Professor Math finds a 3 step process that can best encode the numbers:

1. The first step is to find all possible non-empty subsets of the list of numbers and then, for each subset, find the difference between the largest and smallest numbers (that is, the largest minus the smallest) in that subset. Note that if there is only one number in a subset, it is both the largest and the smallest number in that subset. The complete set itself is also considered a subset.
2. Then add up all the differences to get the final encoded number.
3. As the number may be large, output the number modulo $10^9 + 7$ (1000000007).

The professor has shared an example and its explanation below. Given a list of numbers, can you help the professor build an efficient function to compute the final encoded number?

Input

The first line of the input gives the number of test cases, **T**. This is followed by T test cases where each test case is defined by 2 lines:

1. The first line gives a positive number **N**: the number of numbers in the list and
2. The second line contains a list of **N** positive integers **K_i**, sorted in non-decreasing order.

Output

For each test case, output one line containing `Case #x: y`, where *x* is the test case number (starting from 1) and *y* is the final encoded number.

Since the output can be a really big number, we only ask you to output the remainder of dividing the result by the prime $10^9 + 7$ (1000000007).

Limits

$$1 \leq T \leq 25.$$

Time limit: 20 seconds per test set.

Memory limit: 1GB.

$$1 \leq K_i \leq 10000, \text{ for all } i.$$

$$K_i \leq K_{i+1}, \text{ for all } i < N - 1.$$

Small dataset (Test set 1 - Visible)

$$1 \leq N \leq 10.$$

Large dataset (Test set 2 - Hidden)

$$1 \leq N \leq 10000.$$

Sample

Sample Input

```
1
4
3 6 7 9
```

Sample Output

Case #1: 44

Explanation for the sample input

1. Find all subsets and get the difference between largest & smallest numbers:

[3], largest-smallest = $3 - 3 = 0$.
[6], largest-smallest = $6 - 6 = 0$.
[7], largest-smallest = $7 - 7 = 0$.
[9], largest-smallest = $9 - 9 = 0$.
[3, 6], largest-smallest = $6 - 3 = 3$.
[3, 7], largest-smallest = $7 - 3 = 4$.
[3, 9], largest-smallest = $9 - 3 = 6$.
[6, 7], largest-smallest = $7 - 6 = 1$.
[6, 9], largest-smallest = $9 - 6 = 3$.
[7, 9], largest-smallest = $9 - 7 = 2$.
[3, 6, 7], largest-smallest = $7 - 3 = 4$.
[3, 6, 9], largest-smallest = $9 - 3 = 6$.
[3, 7, 9], largest-smallest = $9 - 3 = 6$.
[6, 7, 9], largest-smallest = $9 - 6 = 3$.
[3, 6, 7, 9], largest-smallest = $9 - 3 = 6$.

2. Find the sum of the differences calculated in the previous step:

$3+4+6+1+3+2+4+6+6+3+6$
 $= 44$.

3. Find the answer modulo $10^9 + 7$ (1000000007):

$44 \% 1000000007 = 44$