Problem B. Assembly via Minimums

Time limit 2000 ms
Mem limit 262144 kB

Sasha has an array a of n integers. He got bored and for all i, j ($i < j$), he wrote down the minimum value of a_i and a_j. He obtained a new array b of size $\frac{n \cdot (n-1)}{2}$.

For example, if $a = [2, 3, 5, 1]$, he would write [
$\min(2, 3), \min(2, 5), \min(2, 1), \min(3, 5), \min(3, 1), min(5, 1)] = [2, 2, 1, 3, 1, 1]$.

Then, he randomly **shuffled** all the elements of the array b.

Unfortunately, he forgot the array a, and your task is to restore any possible array a from which the array b could have been obtained.

The elements of array a should be in the range $[-10^9, 10^9]$.

Input

The first line contains a single integer t ($1 \le t \le 200$) — the number of test cases.

The first line of each test case contains a single integer n ($2 \le n \le 10^3$) — the length of array a.

The second line of each test case contains $\frac{n \cdot (n-1)}{2}$ integers $b_1, b_2, \ldots, b_{\frac{n \cdot (n-1)}{2}}$ ($-10^9 \le b_i \le 10^9$) — the elements of array b.

It is guaranteed that the sum of n over all tests does not exceed 10^3 and for each array b in the test, there exists an original array.

Output

For each test case, output any possible array a of length n.

Sample 1

Input	Output
5 3 1 3 1 2 10 4 7 5 3 5 3 3 5 2 2 2 2 2 2 2 2 2 2 5 3 0 0 -2 0 -2 0 0 -2 -2	1 3 3 10 10 7 5 3 12 2 2 2 2 2 0 -2 0 3 5

Note

In the first sample, Sasha chose the array $[1, 3, 3]$, then the array b will look like $[\min(a_1, a_2) = 1, \min(a_1, a_3) = 1, \min(a_2, a_3) = 3]$, after shuffling its elements, the array can look like $[1, 3, 1]$.

In the second sample, there is only one pair, so the array $[10, 10]$ is suitable. Another suitable array could be $[15, 10]$.