

A problem of Backtracking

You have to solve the following problem with Backtracking. You're given a sequence of **10** positive integers $n_1, n_2, n_3, \dots, n_9, n_{10}$ and a positive value **K**.

To solve this problem you need to print a permutation $a_1, a_2, a_3, \dots, a_{10}$ of the numbers $\{0,1,2,3,4,5,6,7,8,9\}$ such that $a_1 * n_1 + a_2 * n_2 + a_3 * n_3 + \dots + a_{10} * n_{10} \leq K$

Input

In the first line, a single integer T, the number of test cases.

For each test case there will be two lines:

In the first one, 10 positive integers ($1 \leq n_i \leq 10^9$) separated by spaces.

In the second one, a single positive integer K ($1 \leq K \leq 10^9$).

Output

For each test case print a line with the answer for that test case as following:

Among all the permutations that solve the problem according to the description above, print the lexicographically smallest.

You've to print the permutation in a single line, separating each integer by a simple space.

If no such permutation exists, print a single line with "-1".

Example

Input:

```
2
1 2 3 4 5 6 7 8 9 10
200
1 2 3 4 5 6 7 8 9 10
100
```

Output:

```
2 6 8 9 7 5 4 3 1 0
-1
```