D. Find the Different Ones!

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given an array a of n integers, and q queries.

Each query is represented by two integers l and r ($1 \le l \le r \le n$). Your task is to find, for each query, two indices i and j (or determine that they do not exist) such that:

- $l \le i \le r$;
- $l \le j \le r$;
- $a_i \neq a_j$.

In other words, for each query, you need to find a pair of different elements among $a_l, a_{l+1}, \ldots, a_r$, or report that such a pair does not exist.

Input

The first line of the input contains a single integer t ($1 \le t \le 10^4$) — the number of test cases. The descriptions of the test cases follow.

The first line of each test case contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the length of the array a.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^6$) — the elements of the array a.

The third line of each test case contains a single integer q ($1 \le q \le 2 \cdot 10^5$) — the number of queries.

The next q lines contain two integers each, l and r ($1 \le l < r \le n$) — the boundaries of the query.

It is guaranteed that the sum of the values of n across all test cases does not exceed $2 \cdot 10^5$. Similarly, it is guaranteed that the sum of the values of q across all test cases does not exceed $2 \cdot 10^5$.

Output

For each query, output two integers separated by space: i and j ($l \le i, j \le r$), for which $a_i \neq a_j$. If such a pair does not exist, output $i = -1$ and $j = -1$.

You may separate the outputs for the test cases with empty lines. This is not a mandatory requirement.

Example

input Copy

```
5
5
1 1 2 1 1
3
1 5
1 2
1 3
6
30 20 20 10 10 20
5
1 2
2 3
2 4
2 6
3 5
4
5 2 3 4
4
1 2
```

```
1 4
2 3
2 4
5
1 4 3 2 4
5
1 5
2 4
3 4
3 5
4 5
5
2 3 1 4 2
7
1 2
1 4
1 5
2 4
2 5
3 5
4 5
```

output Copy

```
2 3
-1 -1
1 3

2 1
-1 -1
4 2
4 6
5 3

1 2
1 2
2 3
3 2

1 3
2 4
3 4
5 3
5 4

1 2
4 2
1 3
2 3
3 2
5 4
5 4
```