Problem C

Counting Partition

You are given two arrays of N integers, $A_{1..N}$ and $B_{1..N}$, and Q queries of $\langle X, Y\rangle$. For each query, you are asked to split array A into a number of partitions such that in each partition except the last one:

1. Contains at least B_Y elements,
2. Integer X appears at least Y times,
3. The last element is an integer X.

The only requirement for the last partition is that it must contain at least one element; it doesn't need to satisfy the above requirements.

Your task is to find the maximum number of partitions which can be made for each query.

For example, let $A[1..6]$ = {1, 1, 2, 2, 1, 1}, $B[1..6]$ = {1, 3, 3, 4, 5, 6}, and you are given 3 queries:

- Query $\langle 1, 1\rangle$. We can split A into 4 partitions in this query, e.g., {1 | 1 | 2, 2, 1 | 1}, such that each partition (except the last) contains at least $B_{Y=1}$ = 1 element, the integer X = 1 appears at least Y = 1 time, its last element is X = 1, and this is the largest number of partitions which can be made.
- Query $\langle 1, 2\rangle$. We can split A into 2 partitions in this query, e.g., {1, 1, 2, 2, 1 | 1}, and this is maximum. Note that partition {1, 1 | 2, 2, 1, 1} is not valid as the first partition should contain at least B_2 = 3 elements.
- Query $\langle 2, 3\rangle$. In this query, each partition except the last one, 2 should appear 3 times. As there are only two 2, then we cannot split A and make the only become the last partition, i.e. {1, 1, 2, 2, 1, 1}.

Input

The first line of input contains T (T ≤ 20) denoting the number of cases. Each case begins with two integers: N (1 ≤ N ≤ 100,000), and Q (1 ≤ Q ≤ 100,000), representing the size of array and the number of query, respectively. The second line of each case contains N integers: A_i (1 ≤ A_i ≤ 10^9) representing array A. The third line of each case contains N integers: B_i (1 ≤ B_i ≤ N) representing array B. The next Q lines, each contains two integers X (1 ≤ X ≤ 10^9) and Y (1 ≤ Y ≤ N) representing the query.

Output

For each case, output "`Case #X:`" (without quotes) in a line where `X` is the case number (starts from 1). For the next Q lines in each case, output the maximum number of partitions for the respective query in a single line (as in input order).

Warning: large input and output file.

Sample Input	Output for Sample Input
3 6 3 1 1 2 2 1 1 1 3 3 4 5 6 1 1 1 2 2 3 4 3 3 1 3 3 1 1 1 1 1 1 2 1 3 1 10 3 1 1 1 1 1 1 1 1 1 1 2 9 1 10 6 2 1 8 1 7 1 2 1 9 2 1	Case #1: 4 2 1 Case #2: 2 1 3 Case #3: 2 2 1

Explanation for 1st sample case

This is the example from the problem statement

Explanation for 2nd sample case

- Query $\langle 1, 1 \rangle$. A can be split into {3, 1 \| 3, 3}. The first partition contains at least B_1 = 1 element, integer 1 appears at least 1 times, and the last element is 1. Second partition is the last partition.
- Query $\langle 2, 1 \rangle$. A cannot be split as there is no integer 2 in array A, whilst this query requires us to have partitions (except the last) in which integer 2 appears at least 1 time in each of them.
- Query $\langle 3, 1 \rangle$. A can be split into {3 \| 1, 3 \| 3}. In both 1st and 2nd partitions, there are at least B_1 = 1 element, integer 3 appears at least 1 times, and the last element is 3. Third partition is the last partition.

Explanation for 3rd sample case

- Query $\langle 1, 2 \rangle$. A can be split into {1, 1, 1, 1, 1, 1, 1, 1, 1 \| 1}. The first partition contains at least B_2 = 9 elements, integer 1 appears at least 2 times, and the last element is 1. Second partition is the last partition.
- Query $\langle 1, 9 \rangle$. A can be split into {1, 1, 1, 1, 1, 1, 1, 1, 1 \| 1}. The first partition contains at least B_9 = 1 element, integer 1 appears at least 9 times, and the last element is 1. Second partition is the last partition.
- Query $\langle 2, 1 \rangle$. There's no integer 2 in A, thus we can only make 1 partition.