D. Seraphim the Owl

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The guys lined up in a queue of n people, starting with person number $i = 1$, to ask Serafim the Owl about the meaning of life. Unfortunately, Kirill was very busy writing the legend for this problem, so he arrived a little later and stood at the end of the line after the n-th person. Kirill is completely dissatisfied with this situation, so he decided to bribe some people ahead of him.

For the i-th person in the queue, Kirill knows two values: a_i and b_i. If at the moment Kirill is standing at position i, then he can choose any position j such that $j < i$ and exchange places with the person at position j. In this case, Kirill will have to pay him a_j coins. And for each k such that $j < k < i$, Kirill will have to pay b_k coins to the person at position k. Kirill can perform this action any number of times.

Kirill is thrifty, so he wants to spend as few coins as possible, but he doesn't want to wait too long, so Kirill believes he should be among the first m people in line.

Help Kirill determine the minimum number of coins he will have to spend in order to not wait too long.

Input

Each test consists of several sets of input data. The first line contains a single integer t ($1 \le t \le 10^4$) — the number of test cases. Then follows the description of the test case.

The first line of each test case contains two integers n and m ($1 \le m \le n \le 200\,000$) — the number of people in the queue besides Kirill and the maximum allowable final position of Kirill, respectively.

The second line contains n integers $a_1, a_2, \ldots, a_n$ separated by spaces ($1 \le a_i \le 10^9$).

The third line contains n integers $b_1, b_2, \ldots, b_n$ separated by spaces ($1 \le b_i \le 10^9$).

It is guaranteed that the sum of the values of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output a single integer — the minimum number of coins Kirill needs to spend.

Example

input

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

output

```
14
22
9
3
```