



Tricks of the Trade (trade)

Since your career as an art thief turned out to be decidedly less glamorous than you had hoped for, you have set your eyes on a new target: you want to steal a medal at this year's CEOI!* So far your evil plan is working as intended: The Scientific Committee, still confused about the oil shortage you had arranged, immediately fell for your feigned hacking attempt. And when they let you into their secret headquarters to help plan the rejudging, you could easily steal the combination for the safe in which the medals are stored...

However, the next phase of your plan will require an assistant, and right now you don't have enough money to hire one. Fortunately, you just found an opportunity to fix this problem. A shop you stumbled upon in Berlin sells N programmable vacuum cleaner robots, numbered from 1 to N . Robot i costs c_i euros. Unfortunately, the vendor will only sell you a full, contiguous segment of robots: If you want to buy robots i and j , then you must buy every robot k with $i \leq k \leq j$.

As your fellow contestants are quite fond of robots, you promised to sell them K of these robots ($1 \leq K \leq N$). They will pay you s_i euros for robot i , and you hope to make a good profit in the process.

Write a program that

- ▶ computes the maximum profit you can make by buying a segment of at least K robots and selling exactly K robots of this segment to the other contestants, *and*
- ▶ determines which robots you might sell to the other contestants as part of such a transaction with maximum profit.†

Input

The first line of input contains the numbers N and K as described above. The second line contains the N integers c_1, \dots, c_N . Finally, the third line contains the N integers s_1, \dots, s_N .

Output

Your program should output two lines. The first line should contain a single integer, the maximum profit you can achieve. On the second line, you should output a binary string of length N : the i -th character should be 1 if you might sell the i -th robot in some transaction with maximum profit and 0 otherwise.

Constraints and grading

We always have $1 \leq N \leq 250\,000$ and $1 \leq c_i, s_i \leq 10^9$.

Subtask 1 (5 + 5 points). $N \leq 200$

Subtask 2 (5 + 5 points). $N \leq 6\,000$

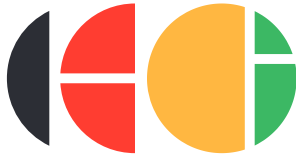
Subtask 3 (5 + 5 points). $K = 2$

Subtask 4 (10 + 15 points). $K \leq 200$

Subtask 5 (25 + 20 points). No further constraints

* Obviously, the problem was with the art, not the thefts.

† After all, you might find some use for robots that you keep...



Partial scoring. In all subtasks, you can obtain partial scores as indicated in the brackets:

- ▶ The first number represents the number of points awarded if your program computes the maximum profit correctly.
- ▶ The second number represents the number of additional points awarded if your program computes the entire output correctly.

Examples

Input	Output
5 3 3 5 2 3 6 2 1 5 2 3	-1 00111
5 2 1 6 1 5 2 4 1 6 2 4	2 10111

In the first example, you can buy the third to fifth robot and sell all three of them to your fellow contestants. This costs you $2 + 3 + 6 = 11$ euros, while the contestants will only pay you $5 + 2 + 3 = 10$ euros for these robots, so you lose 1 euro. If you buy any other segment of robots, you would lose even more money.

In the second example, you can buy the first to third robot and sell the first and third robot to your fellow contestants. This costs you 8 euros, while the contestants will pay you 10 euros, so your profit is 2 euros. No other segment of robots would give you a higher profit. However, you could also buy the third and fourth robot, and sell both of them, or buy the third to fifth robot, and sell the third and fifth robot. In both cases, your profit would also be 2 euros, and so all robots except for the second robot can be part of a transaction with maximum profit.

Limits

Time: 7 s

Memory: 2048 MiB