Building Blocks

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

You are playing with building blocks and have piled up n castles in a row, numbered $1, 2, \ldots, n$ from left to right. The height of the i-th castle is h_i. Now you can perform each of the following two operations any times you want:

- Choose an integer i ($1 \leq i \leq n$) and put some building blocks on the i-th castle to increase its height by 1, which will cost you a_i seconds.
- Choose an integer i ($1 \leq i \leq n$) and remove some building blocks from the i-th castle to decrease its height by 1, which will cost you d_i seconds. Note that the height of a castle can not be negative.

Your goal is to make the sequence $h_1, h_2, \ldots, h_n$ non-decreasing (i.e., $h_i \leq h_{i+1}$ for all $1 \leq i < n$). Please calculate the minimum total seconds you need.

Input

The first line contains an integer T ($1 \leq T \leq 10^5$) — the number of test cases.

For each test case:

- The first line contains an interger n ($1 \leq n \leq 10^5$) — the number of castles.
- The second line contains n integers $h_1, h_2, \ldots, h_n$ ($1 \leq h_i \leq 10^8$) — the height of each castle.
- The third line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^5$) — the seconds you need to increase the height of the i-th castle by 1.
- The fourth line contains n integers $d_1, d_2, \ldots, d_n$ ($1 \leq d_i \leq 10^5$) — the seconds you need to decrease the height of the i-th castle by 1.

It's guaranteed that the sum of n among all test cases will not exceed 10^5.

Output

For each test case, print the minimum seconds you need in a separate line.

Example

standard input	standard output
2 3 3 2 1 3 2 1 1 2 3 10 14 3 4 1 7 18 11 3 8 3 18 19 20 3 17 8 14 18 19 8 7 12 20 5 10 16 17 6 20 8	2 427