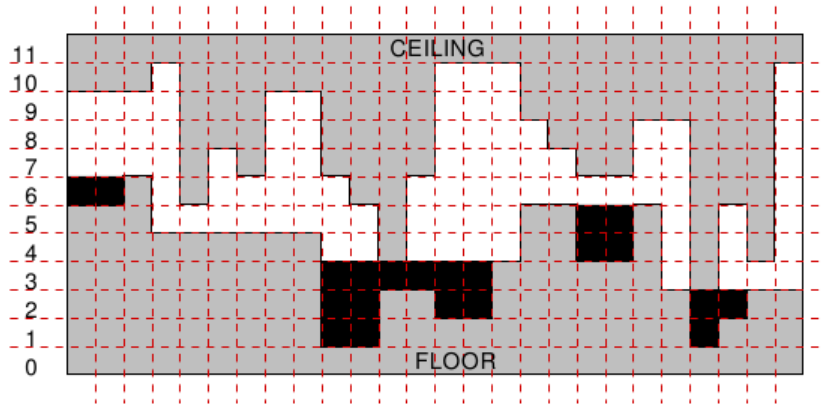
As an owner of a land with a cave you were delighted when you last heard that underground fuel tanks are great business. Of course, the more volume one can store, the better. In case of your cave, the efflective volume is not easy to calculate, because the cave has a rather sophisticated shape (see figure). Thank heavens it is degenerate in one dimension!



The cave. All ponds that can be flooded with fuel are marked black.

Furthermore, there is some electrical wiring on the ceiling of the cave. You can never be sure if the insulation is intact, so you want to keep the fuel level just below the ceiling at every point. You can pump the fuel to whatever spots in the cave you choose, possibly creating several ponds. Bear in mind though that the fuel is a liquid, so it minimises its gravitational energy, e.g., it will run evenly in every direction on a flat horizontal surface, pour down whenever possible, obey the rule of communicating vessels, etc. As the cave is degenerate and you can make the space between the fuel level and the ceiling arbitrarily small, you actually want to calculate the maximum possible area of ponds that satisfy aforementioned rules.

Input

The input contains several test cases. The first line of the input contains a positive integer $Z \leq 15$, denoting the number of test cases. Then Z test cases follow, each conforming to the format described below.

In the first line of an input instance, there is an integer n ($1 \leq n \leq 10^6$) denoting the width of the cave. The second line of input consists of n integers $p_1, p_2, \ldots, p_n$ and the third line consists of n integers $s_1, s_2, \ldots, s_n$, separated by single spaces. The numbers p_i and s_i satisfy $0 \leq p_i < s_i \leq 1000$ and denote the floor and ceiling level at interval $[i, i+1)$, respectively.

Output

For each test case, your program is to print out one integer: the maximum total area of admissible ponds in the cave.

Sample Input

```
1
15
6 6 7 5 5 5 5 5 5 1 1 3 3 2 2
10 10 10 11 6 8 7 10 10 7 6 4 7 11 11
```

Sample Output

```
14
```