# An easy math problem

### Problem Statement

Brother HeZai is a good student, he is good at number theory! He thinks any problem can be solved by DTFT(离散傅里叶变换),but you don’t think so and you give Brother HeZai an “easy” math problem.

Problem’s contents are as follows.

Fast Fourier Transformation is an algorithm used to calculate convolution. Specifically, if $a$, $b$ and $c$ are sequences with length $N$, which are indexed from 0 to $N$ - 1

$$C_i = \sum_{j=0}^{i} a_j * b_{i-j}$$

And we can calculate $c$ fast using Fast Fourier Transformation.

You made a little change on this formula. Now

$$C_i = \max_{0 \le j \le i} a_j * b_{i-j}$$

To make things easier, $a$ is a permutation of integers from $1$ to $N$, and $b$ is a sequence only containing $0$ and $1$. Given $a$ and $b$, Brother HeZai needs to calculate $c$.

Brother HeZai is so powerful! He solves it immediately! But now it’s your turn, can you solve it?

### Input Format

The first line only contains an positive integer T ( T <= 22 ) , represents there are T test cases.

For each test case:

The first line contain one integer N, indicating the length of $a$. $1 \le \mathrm{N} \le 10^4$

The next line contains a permutation from 1 to N, indicating the elements of $a$.

The next line contains a 0-1 sequence of length N, indicating the elements of $b$.

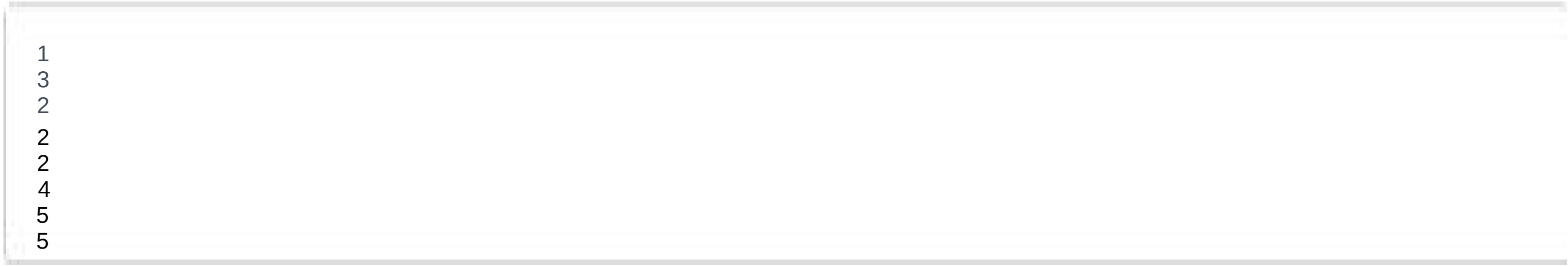
### Output Format

For each test case, output the element of $c$ in N line.

### Sample Input

```
 2
3
1 3 2
1 0 0
5
2 1 4 5 3
1 1 1 0 1
```

### Sample Output



```
1
3
2
2
2
4
5
5
```

### Explanation

For the first test case:
*a* is [1 3 2], *b* is [1 0 0], so $c_0 = max(1 \cdot 1) = 1$, $c_1 = max(1 \cdot 0, 3 \cdot 1) = 3$, $c_2 = max(1 \cdot 0, 3 \cdot 0, 2 \cdot 1) = 2$.