Problem A. V–o_o–V

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

Last Month, Hussain was participating a programming contest. After only 10 minutes he started writing the solution of a problem about trees .

```
vector<long long> answer;
for (int j=2; j<=n; j++){        //Normal c++ loop with j as variable
    long long sum=0;
    for (int i=1; i < j; i++){        //Normal c++ loop with i as variable
        sum += V[ LCA(i,j) ];
        /*
        LCA(i, j) is a predefined function that calculates the
                lowest common ancestor of 2 nodes in a tree
        V[LCA(i, j)] is the value written on that node (LCA).
        */
    }
    answer.push_back(sum);        //Put the answer of i-th node in the answer vector.
}
```

Well, Hussain kept getting TLE. Seems his solution was very slow. He sent over 10 submissions with TLE verdict. He knew that when the contest is finished if the problem isn't solved all his friends will call him a crab.

Noobs in gaming are called crabs (because they have claws not normal fingers) that's why they are bad at gaming (actually you can call noob coders crabs the same way).

Hussain asked you for help, can you write a code that performs exactly the same objective and write the answer vector to output for all test-cases (please see the example for better understanding).

Input

The first line contains a single integer T, the number of test cases.

Each test case starts with a line containing a single integer N the number of nodes ($2 \leq N \leq 2 * 10^5$).

The next line contains N space separated integers. The i_{th} integer represents V_i which is the value written on the i_{th} node. ($1 \leq V_i \leq 10000$)

The next line contains N space separated integers. The i_{th} integer represents P_i which is the parent of the i_{th} node. Plase note that $P_1 = 0$ always. You can assume that the first node is the root of the tree.

It's guaranteed that the input is a valid tree.

Output

For each test-case output a single line consisting of $N - 1$ integers. The i_{th} of them represents the sum associated with the $(i + 1)_{th}$ node. In fact you need to output the vector "answer"in the source code as it's exactly.

Example

standard input	standard output
2 5 1 2 3 4 5 0 1 2 3 4 3 100 100 100 0 1 1	1 3 6 10 100 200