Problem K. Tree generation

Input file:	standard input
Output file:	standard output
Time limit:	2 seconds
Memory limit:	256 megabytes

After the last Codeforces round Vasya has become obsessed with creation of his own problems. Vasya has a poor imagination, but he hates cactuses, so he decided to create a problem about tree. He easily came up with just another task where you need to run two dfs on the tree, but the next problem he encountered is to generate tests for it.

Vasya's most favorite function from testlib.h is wnext. For our purposes, in this problem we simplify its definition, and it works in the following way:

```
int wnext(int a, int b, int type) {
   int result = next(a, b);
   for (int i = 0; i < type; i++)
        result = max(result, next(a, b));
   for (int i = 0; i < -type; i++)
        result = min(result, next(a, b));
   return result;
}
```

Function $next(a, b)$ returns the uniformly distributed random integer in range $[a; b]$.

Vasya came up with the following way to generate tests: he generates tree with $n = 10000$ vertices. Vertices of the tree are numbered from 0 to $n - 1$. He chooses random integer *type* in range $[-4; 4]$, iterates over vertices of the tree from vertex 1 to vertex $n-1$, adding an edge between vertex i and vertex $wnext(0, i - 1, type)$. After that, Vasya randomly shuffles the vertices of the tree. This procedure can be described by the following code:

```
int n = 10000;
int p[n];
for (int i = 0; i < n; ++i) {
    p[i] = i;
}
shuffle(p, p + n);
int type = next(-max_param, max_param);
for (int i = 1; i < n; ++i) {
    add_edge(p[i], p[wnext(0, i - 1, type)]);
}
```

Here p is a random permutation of integers from 0 to $n - 1$.

Vasya generated 100 testcases for his problem in such way. Accidentally, he forgot to mark each testcase by the value of the parameter *type* used in it. Given the testcases generated by Vasya, help him to recover the parameter values for at least 90% of testcases. Parameters for all testcases are choosen independently.

Input

The first line contains one integer T — the number of testcases, which is always equal 100.

Next T lines contain the decsription of testcases. Each line starts with integer n — the number of vertices in the tree, which is always equal 10000. The next $n - 1$ intgers $p_1, p_2, \ldots, p_{n-1}$ $(0 \le p_i \le n - 1, p_i \ne i)$ denote the edges of the tree (i, p_i). Note that the vertices were shuffled by Vasya, so it's not guaranteed that $p_i < i$. It's guaranteed that edges (i, p_i) form a tree on n vertices.

In the sample, just to show the format, $T = 2$, $n = 3$. The sample was not generated as shown above, and your solution will not be tested on the sample. The first test is different from the sample.

Output

Output T integers, one per line, — the values of the parameter *type*, used to generate testcases. To get OK for the test, you should print them correctly for at least 90 testcases.

Examples

standard input	standard output
2 3 0 1 3 0 0	-2 1