I – Interactive Reconstruction

Time limit: 2 s *Memory limit: 256 MiB*

This is an interactive task where your program will communicate with a grader through standard input and output. Your task is to reconstruct a labelled tree with N nodes and $N-1$ edges. Nodes are labelled from 1 to N.

Your program is allowed to make a few queries of the following type: Your program should print a string of N characters, consisting only of zeros and ones, one corresponding to each node. The grader will return a sequence of N space-separated integers, the i-th representing the sum of the values (i.e. digits of the query string) of all neighbours of the i-th node. That is, if node j is a neighbour of node i, then the j-th digit of the query string counts towards the sum in the i-th number of the grader's answer.

See the example below for an illustration.

Input and output data

The first input line will contain the number N, the number of nodes in the tree.

Your program then has two options:

1. Print "`QUERY`" (without quotation marks), a space, and a string of N zeros and ones.
2. Print "`ANSWER`" (without quotation marks), a newline, and $N-1$ lines, each containing a pair of space-separated integers a, b, indicating that there exists an edge between nodes a and b.

If your program prints a query, this will be followed by the grader returning a line with N space-separated integers, one per node. If your program prints an answer, the grader will check the returned tree for correctness.

If there was a mistake in your queries, either due to incorrect formatting or due to an exceeded number of queries, the grader will print "`ERROR`" (no quotation marks) instead of the answer.

Important: Ensure that your program flushes its output after printing and correctly exits after printing the answer. It is up to you whether to implement the `ERROR` handling. Its purpose is to allow your program to exit gracefully and get a WA instead of a TLE verdict in the case of an error.

Input limits

- $2 \leq N \leq 3 \cdot 10^4$
- At most $2 \uparrow\uparrow 3 = 2^{(2^2)} = 16$ queries are allowed. The final answer does not count toward this restriction.

Example

Program output	Grader output
	`5`
`QUERY 10001`	
	`0 0 1 2 0`
`QUERY 00010`	
	`1 1 0 0 1`
`QUERY 10000`	
	`0 0 0 1 0`

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

Comment

The tree in question is the following one:

```
1-4-2
  |
  5-3
```

With the three queries in the example, it is possible to reconstruct it uniquely.