A. cbfgweiuhui-pt-1

time limit per test: 15 seconds
memory limit per test: 1024 megabytes

Santa Claus sees the world as a d-dimensional hypercube, so he stores the addresses of kids around the globe as a set of n d-dimensional binary vectors. This year, he prepared an Approximate Near Neighbour Search (ANNS) data structure on this dataset, and he is planning to use it for delivering presents during Christmas night! Santa's memory is bit rusty, so he wants that if he lands with his sleigh at a point q on the hypercube that is at distance at most r from some kid's house, then he can interrogate the ANNS data structure and find some kid at distance at most $c \cdot r$ from q with probability $1 - 1/n$; if instead he lands at a point q that is at distance larger than $c \cdot r$ from all kids, his data structure will tell him that no kid lives close by and he will fly somewhere else.

As the Grinch, your role is of course that of ruining Christmas! Assume that there is a kid that lives at a location z that is at distance at least $2 \cdot \lceil c \cdot r \rceil + 2$ from every other kid. You want to find a point q at distance at most r from z such that if Santa lands at q, his data structure will make him believe that there is no kid living within distance $c \cdot r$. Hence, if Santa lands at q he will be close to z and yet he will leave the poor kid living there with no present! The only constraint is that you can only interrogate the data structure at most N times, otherwise the IT elves will notice the suspicious network traffic.

Input

The first line of the input contains the following space-separated values: d, r, c, n, N. The dimension d is an integer $10 \leq d \leq 1000$. The ANNS parameter r is an integer $2 \leq r \leq 200$. The ANNS approximation c is a float $1.5 \leq c \leq 10$. The number of points in the dataset n is an integer $3 \leq n \leq 10000$. The maximum number of allowed queries N is an integer $40 \leq N \leq 1500$.

The second line of the input contains d space-separated integers in $\{0, 1\}$, corresponding to the location z of the isolated kid in the dataset.

Interaction

A query for a point q is issued by printing to the standard output a single line containing the character `q` followed by a space and then d space-separated integers in $\{0, 1\}$. For example if $d = 5$, a query for the point $q = (1, 0, 0, 1, 0)$ is issued by printing `q 1 0 0 1 0`.

The answer to the query can be read from the standard input. Answers to queries consist of a single line containing an integer m followed by a space and then m space-separated integers. If data structure answers to a query with a point p in the dataset, then $m = d$ and the following integers are the values of the coordinates of p. If instead the data structure does not find a near point in the dataset, then $m = 1$ and the following integer is -1.

After at most N queries as above, the final answer q^* is reported by printing to the standard output a single line containing the character `*` followed by a space and then d space-separated integers in $\{0, 1\}$. For example if $d = 5$, the point $q^* = (1, 0, 0, 1, 0)$ is reported as a solution by printing `* 1 0 0 1 0`. After this, your program should terminate.

After printing a line, make sure to flush the standard output.