Hacking the Project

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 mebibytes

This is an interactive problem.

Lewis is one of the developers of the new programming language called DiverC. The main feature of the program written in this language is that the code consists of pairwise distinct words. The compiler of DiverC developed by Lewis is, of course, written in DiverC and consists of N pairwise distinct words.

Lewis is using the DiverC online autofill service. But Lewis has made one serious mistake: he forgot to switch the "use my data for the improvement of the service database" function off. And Lewis was the first person who registered on this service, so now the service contains only the words from his compiler.

Hacker Fernando wants to know all the words Lewis used in the compiler. So he registered at the DiverC online autofill service (wisely switching the dangerous function off), and now, for each prefix S and integer K entered by Fernando, the service returns, in lexicographic order, the first K words from Lewis's code that begin with the prefix S. If there are only $k < K$ words, the service gives out only k words (but the service usage counter increases by K even in this case).

Fernando checked the scripts used for the online service and found that one user is limited with 3 800 as the total value of K in all queries. He wants to determine all N words used by Lewis with several queries such as the sum of K in those queries is 3 800 or less.

Can you help him?

Interaction Protocol

In the beginning, your program shall read one integer T — the number of the test cases to be processed ($1 \leq T \leq 5$).

At the beginning of each test case, the jury program tells one integer N — the number of the words in Lewis's DiverC compiler ($1 \leq N \leq 1\,000$).

Your program can then make two types of requests:

- `query S K` — get K ($1 \leq K \leq N$) lexicographically minimal words starting with prefix S ($1 \leq |S| \leq 10$). If the dictionary contains only k such words, where $k < K$, the answer to the query will contain k words. The response to the query will be one line of the form $kS_1S_2 \ldots S_k$, where k is the number of the words ($0 \leq k \leq K$), and then k words S_i in lexicographic order follow.
- `answer` S_1 S_2 $\ldots S_N$ — tell the full Lewis's dictionary. After the word `answer` you shall print all N words in an arbitrary order separated by spaces. There will be no response from the jury program to this request, and your program must then continue with the next test case or exit if the current test case was the last one.

The words in Lewis's code are composed of lowercase English letters. The length of words is between 1 to 10 characters. All words in Lewis's code are pairwise distinct.

The sum of K for all queries of the first type for each test should be 3 800 or less.

Violating the interaction protocol or exceeding the limits for the sum of K cause the "`Wrong answer`" verdict.

Make sure you print the newline character after each query and flush the output stream buffer (`flush` language command) after each request. Otherwise, the solution may get the idleness limit exceeded verdict.

Note that the jury program is **adaptive**, i.e. the set of Lewis's words may be generated at the runtime, but the set is guaranteed to be consistent with the answers to previous queries.

Example

standard input	standard output
1	
4	
	query a 1
1 aaa	
	query a 4
2 aaa aba	
	query c 1
1 cxyxy	
	query cy 1
0	
	query cz 1
1 czzzz	
	answer aaa aba czzzz cxyxy