Debugging Terence Tao's Code

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

The Fibonacci sequence is a sequence of numbers where each number is the sum of the two preceding ones.

That is, the sequence follows the recursive formula

$$F_1 = 1, F_2 = 1, F_n = F_{n-1} + F_{n-2} \text{ for } n \geq 3.$$

Resulting in:

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \ldots$$

The sequence was discovered by Leonardo of Pisa, a 13th-century mathematician. The great (contemporary) mathematician Terence Tao wrote a program at the age of 6 to calculate all the terms of the sequence less than a given number n. The program is shown in Figure 1.

The program is written in a variant of the BASIC language. It exhibits characteristics of this language, such as the fact that line numbers are written manually (and do not have to be consecutive numbers) and the execution control is done by jumping to indicated line numbers: the instruction `go to N` jumps to line `N`, and the instruction `if ([cond]) then N` jumps to line `N` if condition `cond` is met. Apart from that, the language is similar to other imperative languages like C++, Java, or Python; it is not necessary to know BASIC in detail to solve this problem.

The program has some curious peculiarities and "additional functionalities", but in this problem, we will focus on the part that calculates the Fibonacci numbers, i.e., the numbers that are printed by line 240 of the program. It turns out that, although it seems that the program's objective is to print all Fibonacci numbers (strictly) less than a given number n, it does not always satisfy this objective correctly. Can you tell, given a number n, which Fibonacci number is left unprinted?

Input

The input begins with a number T — the number of test cases.

Next, T lines follow, each with a number n.

Output

For each value of n, you should print a line with "`OK`" (in uppercase and without quotes) if Terence Tao's program correctly prints all Fibonacci numbers (strictly) less than n. Otherwise, you should print a term of the sequence less than n that Tao's program does not print.

Scoring

1. (50 points) $n \leq 100$.
2. (50 points) No additional restrictions.

```
8 print "J"
10 print "here comes mr. fibonacci"
20 print "can you guess which year was mr. fibonacci born?"
30 print "write down a number please . . . ": input c
31 if c = 1170 then print "you are correct! now we start": go to 150
50 if c > 1250 then print "no, he is already in heaven, try again": go to 30
60 if c < 1170 then print "sorry, he wasn't born yet! try again": go to 30
70 if c > 1170 < 1250 then print "he would be ";c-1170;" years old"
71 print "now can you guess?": input c
72 if c = 1170 then 31
73 print "you are wrong. try again.": go to 71
150 print "up to which number do you want me tell you all the fibonacci numbers"
151 input n
160 print "J"
190 print "okay. here they go!"
200 s = 1
210 t = 1
220 if s > = n then 270
230 if t > = n then 270
240 print s; t;
250 s = s + t
260 t = t + s
265 go to 220
270 print
271 print "another game, while fibonacci is waiting (y), or no more? (n)": print
272 get c$: if c$ = " " then 272
273 if c$ = "y" then 150
274 if c$ = "n" then 300
280 go to 272
300 print "mr. fibonacci is leaving now,"
310 print "and wishes to see you again sometime in the future"
312 print
313 print
315 print "here goes his car!!!!!!!!"
320 print "(brmmmm-brmmmm-putt-putt-vraow-chatter-chatter bye mr. fibonacci!)"
390 go to 450
410 print
420 print
445 next i
450 end
```

Рис. 1: Terence Tao's "Fibonacci"program. Source: M.A. Clemens, Educational Studies in Mathematics, Aug, 1984, Vol. 15, No. 3

Example

standard input	standard output
5 2 3 4 5 8	OK 2 OK OK 5

Note

$1 \leq T \leq 1000$.

$2 \leq n \leq 10^8$.