



Depurando Código de Terence Tao

La sucesión de Fibonacci es una sucesión de números donde cada número es la suma de los dos anteriores.

Es decir, la sucesión sigue la fórmula recurrente

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

Dando lugar a:

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

La sucesión fue descubierta por Leonardo de Pisa, un matemático del siglo XIII . El gran matemático (de la actualidad) Terence Tao escribió un programa cuando tenía 6 años para calcular todos los términos de la sucesión menores que un número n dado. El programa se muestra en la Figura 1.

El programa está escrito en una variante del lenguaje BASIC. En él se pueden observar características de este lenguaje, como el hecho de que los números de línea se escriben manualmente (y no tienen por qué ser números consecutivos) y el control de ejecución se realiza mediante saltos a números de línea indicados: la instrucción `go to N` salta a la línea N , y la instrucción `if ([cond]) then N` salta a la línea N si se cumple la condición `cond`. Aparte de eso, el lenguaje es similar a otros lenguajes imperativos como C++, Java o Python; no es necesario conocer BASIC en detalle para resolver este problema.

El programa tiene algunas peculiaridades curiosas y “funcionalidades adicionales”, pero en este problema nos centraremos en la parte que calcula los números de Fibonacci, es decir, los números que son impresos por la línea 240 del programa. Resulta que, aunque parece que el objetivo del programa es imprimir todos los números de Fibonacci (estrictamente) menores que un número n dado, no siempre satisface este objetivo correctamente. ¿Puedes decir, dado un número n , qué número de Fibonacci deja sin imprimir?

Entrada y salida

La entrada comienza con un número T — el número de casos de prueba.

A continuación siguen T líneas, cada una con un número n .

Por cada valor de n , debes imprimir una línea con “OK” (en mayúsculas y sin comillas) en caso de que el programa de Terence Tao imprima correctamente todos los números de Fibonacci (estrictamente) menores que n . En caso contrario, debes imprimir un término de la sucesión menor que n que el programa de Tao no imprima.



```
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
```

Figura 1: Programa "Fibonacci" de Terence Tao. Fuente: M.A. Clemens, Educational Studies in Mathematics, Aug, 1984, Vol. 15, No. 3



Ejemplo

Entrada:

5
2
3
4
5
8

Salida:

OK
2
OK
OK
5

Restricciones

$$1 \leq T \leq 1000.$$

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

Subtareas

1. (50 puntos) $n \leq 100$.
2. (50 puntos) Sin restricciones adicionales.