

Problema número 418

RENUM

Tiempo máximo: 2,000 s Memoria máxima: 4096 KiB

<http://www.aceptaelreto.com/problem/statement.php?id=418>

En los albores de la informática, cuando los entornos de desarrollo estaban por inventar, los lenguajes de alto nivel eran una rareza, la programación funcional estaba naciendo y la programación orientada a objetos por nacer, apareció el lenguaje que marcaría la infancia a toda una generación de informáticos: el BASIC.

Una de las peculiaridades del lenguaje original era la existencia de *números de línea*. Debido a la ausencia de editores de texto para construir el código fuente, cada línea que se deseaba añadir debía ir antecedida del número de línea que ocuparía. Esa numeración (que podía tener huecos entre una línea y la siguiente) se utilizaba también a modo de etiqueta en las instrucciones de salto.

Un ejemplo de programa en BASIC es el siguiente:

```
5 REM Euclid's algorithm for greatest common divisor
6 PRINT "Type two integers greater than 0"
10 INPUT A,B
20 IF B=0 THEN GOTO 80
30 IF SGN(A-B) = 1 THEN GOTO 60
40 LET B=B-A
50 GOTO 20
60 LET A=A-B
70 GOTO 20
80 PRINT A
90 END
```

Como se ve, cada línea de código comienza con el número que la identifica, de forma que otras instrucciones de salto pueden referenciarlas (GOTO xxx o GOSUB xxx).

Durante la edición era habitual que la numeración fuera múltiplos de 10, para, en caso de necesidad, poder añadir líneas de código adicionales cómodamente. Además, llegado el momento, siempre se podía utilizar la orden RENUM para *renumerar* las líneas de nuevo automáticamente.

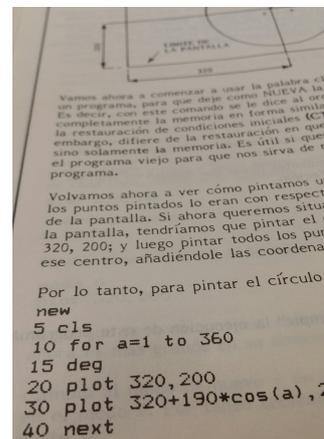
Entrada

La entrada comienza con un entero indicando el número de casos de prueba que vendrán a continuación.

Cada caso de prueba consiste en un programa de como mucho 100 líneas en un BASIC muy simplificado en donde cada línea de código:

- Comienza con el número de línea al que representa. La numeración es ascendente con números positivos entre 1 y 10^9 .
- Tras el número de línea aparece un espacio en blanco.
- A continuación aparece una única instrucción, que podrá ser únicamente GOTO, GOSUB o RETURN. Las dos primeras van seguidas del número de línea al que saltar (con un único espacio de separación) y se garantiza que éste siempre existe.

El listado del programa termina siempre con una línea con numeración 0, que no se considera parte de él.



Salida

Por cada caso de prueba se escribirá el programa en BASIC equivalente tras la renumeración de las líneas.

Esa renumeración asignará el número 10 a la primera línea de código e irá incrementando éste de 10 en 10.

Separa cada listado con una línea con tres guiones (---).

Entrada de ejemplo

```
2
2 GOTO 5
5 GOSUB 50
50 RETURN
0
10 GOSUB 50
11 GOTO 10
50 RETURN
0
```

Salida de ejemplo

```
10 GOTO 20
20 GOSUB 30
30 RETURN
---
10 GOSUB 30
20 GOTO 10
30 RETURN
---
```

Autores: Marco Antonio Gómez Martín y Pedro Pablo Gómez Martín.

Revisor: Alberto Verdejo.