

El problema $3n + 1$

Argumento

Los problemas en Ciencias de la Computación son habitualmente clasificados en ciertos tipos de problemas (p. ej. NP, sin solución, recursivo, etc). En este problema Ud. analizará una propiedad de un algoritmo cuya clasificación no es conocida para todas las posibles entradas.

El problema

Considere el algoritmo siguiente:

```
1.          input  $n$ 
2.          print  $n$ 
3.          if  $n = 1$  then STOP
4.          if  $n$  is odd (impar) then  $n = 3n + 1$ 
5.                               else  $n = n / 2$ 
6.          GOTO 2
```

Ingresando 22 en la entrada, la siguiente secuencia de números será impresa: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

Se ha conjeturado que el algoritmo descripto terminará (cuando se imprima un 1) para cualquier numero entero que se le de como entrada. A pesar de la simplicidad del algoritmo, no se conoce si esta conjetura es cierta. Sin embargo, ha sido verificada para todos los enteros n tales que $0 < n < 1,000,000$ (y, de hecho, para muchos más numeros)

Dada una entrada n , es posible determinar la cantidad de numeros impresos (incluyendo al 1). Para un n dado, esta cantidad es llamada longitud del ciclo (*cycle-length*) de n . En el ejemplo de arriba, la longitud de ciclo de 22 es 16.

Para cualquier par de números i y j Ud. deberá determinar la máxima longitud de ciclo de todos los números encontrados entre i y j .

La entrada

La entrada consistirá en una serie de pares de enteros i y j , cada uno de los pares en una línea. Todos los enteros serán menores a 1.000.000 y mayores a 0.

Ud. deberá procesar todos los pares de enteros y para cada uno de ellos determinar la máxima longitud de ciclo encontrada entre las longitudes de ciclo de los números comprendidos entre i y j inclusive.

La salida

Para cada par de enteros de entrada i y j Ud. deberá presentar: i , j , y la longitud de ciclo máxima. Estos tres números deberán estar separados por al menos un espacio, con los tres números en una misma línea y con una línea de salida por cada línea de entrada. Los enteros i y j deberán aparecer en la salida en el mismo orden en que aparezcan en la entrada.

Ejemplo de entrada

```
1 10
100 200
201 210
900 1000
```

Ejemplo de salida

```
1 10 20
100 200 125
201 210 89
900 1000 174
```