

Escenario

En 1937, un matemático alemán llamado Lothar Collatz formuló una hipótesis intrigante (aún no se ha comprobado) que se puede describir de la siguiente manera:

1. Toma cualquier número entero que no sea negativo y que no sea cero y asígnale el nombre `c0`.
2. Si es par, evalúa un nuevo `c0` como `c0 ÷ 2`.
3. De lo contrario, si es impar, evalúa un nuevo `c0` como `3 × c0 + 1`.
4. Si `c0 ≠ 1`, salta al punto 2.

La hipótesis dice que, independientemente del valor inicial de `c0`, el valor siempre tiende a 1.

Por supuesto, es una tarea extremadamente compleja usar una computadora para probar la hipótesis de cualquier número natural (incluso puede requerir inteligencia artificial), pero puede usar Python para verificar algunos números individuales. Tal vez incluso encuentres el que refutaría la hipótesis.

Escribe un programa que lea un número natural y ejecute los pasos anteriores siempre que `c0` sea diferente de 1. También queremos que cuente los pasos necesarios para lograr el objetivo. Tu código también debe mostrar todos los valores intermedios de `c0`.

Sugerencia: la parte más importante del problema es como transformar la idea de Collatz en un ciclo `while`- esta es la clave del éxito.

Prueba tu código con los datos que hemos proporcionado.

Datos de prueba

Entrada de muestra: `15`

Salida esperada:

```
46
23
70
35
106
53
160
```

80
40
20
10
5
16
8
4
2
1
pasos = 17

Entrada de muestra: 16

Salida esperada:

8
4
2
1
pasos = 4

Entrada de muestra: 1023

Salida esperada:

3070
1535
4606
2303
6910
3455
10366
5183
15550
7775
23326
11663
34990
17495
52486
26243
78730
39365
118096

59048
29524
14762
7381
22144
11072
5536
2768
1384
692
346
173
520
260
130
65
196
98
49
148
74
37
112
56
28
14
7
22
11
34
17
52
26
13
40
20
10
5
16
8
4
2
1
pasos = 62