

---

## Secuencia alícuota

X30687\_es

---

La sucesión alícuota de un número entero  $n$  comienza con el propio número  $n$  seguido de términos que son la suma de los divisores propios del término anterior.

Por ejemplo, la secuencia alícuota de 10 es la secuencia: 10, 8, 7, 1 dado que:

```
n = 10, primer valor
[10]
divisores propios de 10: 5 + 2 + 1 = 8, segundo valor
[10, 8]
divisores propios de 8: 4 + 2 + 1 = 7, tercer valor
[10, 8, 7]
divisores propios de 7: 1, cuarto valor
[10, 8, 7, 1]
divisores propios de 1: no tiene, testigo de fin.
[10, 8, 7, 1]
```

Hay valores para los cuáles la secuencia alícuota es infinita, por ejemplo, para  $n = 6$ . De hecho, siempre que un término ya forme parte de la secuencia, ésta será infinita. Por ejemplo: 95, 25, 6, 6, 6 ... que daría como resultado la lista [95, 25, 6, 6] ya que al repetirse el 6 ya no hace falta seguir calculando más números.

La función *divisores\_propios(n)* ofrecida a continuación retorna la lista de divisores propios de un entero, cópiala y utilízala en la solución del problema:

```
def divisores_propios(n):
    '''
    n >= 1
    Devuelve la lista con los divisores propios de n
    Si n no tiene ningun divisor propio, devuelve la lista vacia

    >>> divisores_propios(10)
    [1, 2, 5]
    >>> divisores_propios(6)
    [1, 2, 3]
    >>> divisores_propios(1)
    []
    '''
    if n == 1:
        return []
    result = [1]
    d = 2
    while d*d <= n:
        if n%d == 0:
            result.append(d)
            if n//d != d:
                result.append(n//d)
        d += 1
    return result
```

Diseñad una función *secuencia\_alicuota(n, maxim)*, que dados dos enteros *n* y *maxim* mayores que cero, devuelva una lista con la secuencia alícuota de *n* hasta encontrar un testigo. El testigo puede ser o bien uno, o bien el primer término que supera a *maxim*, o bien el primer término que se repite. Este testigo se incluirá en la lista resultado como último elemento. La función *secuencia\_alicuota(n, maxim)* tiene que utilizar la función *divisores\_propios(n)*.

### Ejemplo de sesión

```
>>> secuencia_alicuota(12, 25)
[12, 16, 15, 9, 4, 3, 1]
>>> secuencia_alicuota(6, 10)
[6, 6]
>>> secuencia_alicuota(95, 100)
[95, 25, 6, 6]
>>> secuencia_alicuota(102, 300)
[102, 114, 126, 186, 198, 270, 450]
>>> secuencia_alicuota(102, 800)
[102, 114, 126, 186, 198, 270, 450, 759, 393, 135, 105, 87, 33, 15, 9, 4, 3, 1]
```

### Información del problema

Autor : InfBesos

Generación : 2017-11-16 19:57:24

© *Jutge.org*, 2006–2017.

<http://jutge.org>