

# Sucesión de naturales

Dados dos naturales  $x$  e  $y$ , a continuación se define una sucesión de naturales:

$$\begin{aligned}fob_0(x, y) &= x \\fob_1(x, y) &= y \\fob_n(x, y) &= fob_{n-1}(x, y) + fob_{n-2}(x, y) \text{ si } n \geq 2\end{aligned}$$

Se pide:

- Diseñar un algoritmo recursivo lineal no final que calcule  $fob_n(x, y)$ .
  - Diseñar un algoritmo recursivo lineal final que calcule  $fob_n(x, y)$ .
- Ambos algoritmos deben tener coste en  $O(n)$

## Entrada

La entrada consta de varias líneas cada una de las cuales contiene: o bien un -1 para indicar que ya no hay más casos de prueba o bien los datos  $n$ ,  $x$  e  $y$  de un caso de prueba.

## Salida

El resultado de cada caso de prueba aparecerá en una línea, como se muestra en el ejemplo.

## Entrada de ejemplo

```
0 5 6
1 3 4
2 9 5
15 10 24
-1
```

## Salida de ejemplo

```
5
4
14
18410
```

## Nota

Este ejercicio debe verse en el contexto de la asignatura de Estructura de Datos y Algoritmos (EDA), FDI-UCM 2016/2017 (prof. Clara Maria Segura Diaz). Por tanto *no* vale cualquier solución, sino sólo aquellas que utilicen los conceptos de EDA. Es muy posible que se den aclaraciones adicionales en clase a este respecto.