

Actividad refuerzo SOLUCIÓN

🔨 AAR2.17 Completa

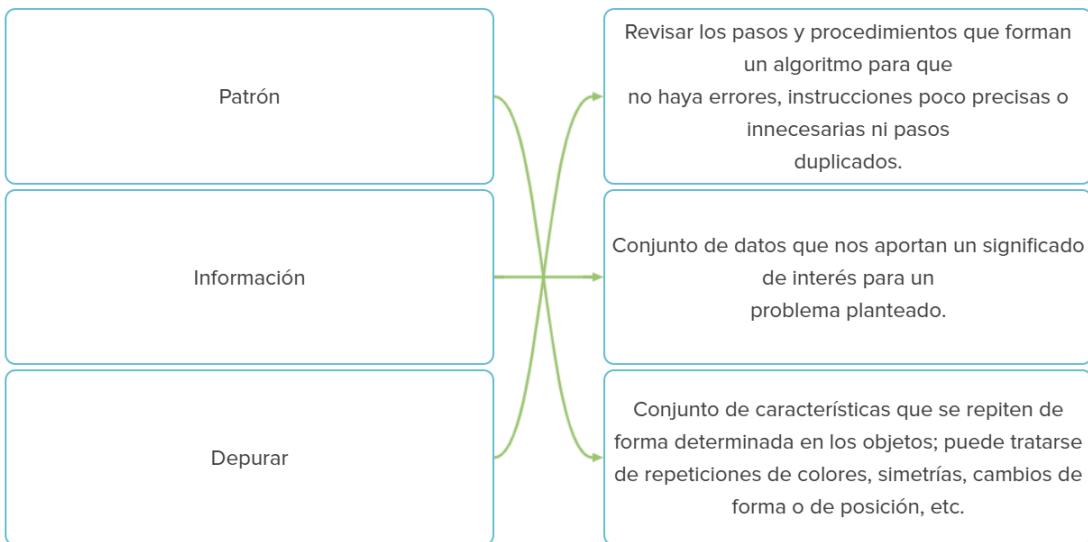
(C.ESP2 / CE2.2, CE2.3, CE2.4 / IC2-10p)

- a) En ocasiones, trabajar como lo haría un ordenador nos puede ayudar a resolver problemas de la vida diaria y desarrollar soluciones sencillas y creativas para problemas habituales.
- b) Esta forma de trabajo se denomina pensamiento computacional y consiste en abordar los problemas planteados como si fuéramos científicos informáticos.

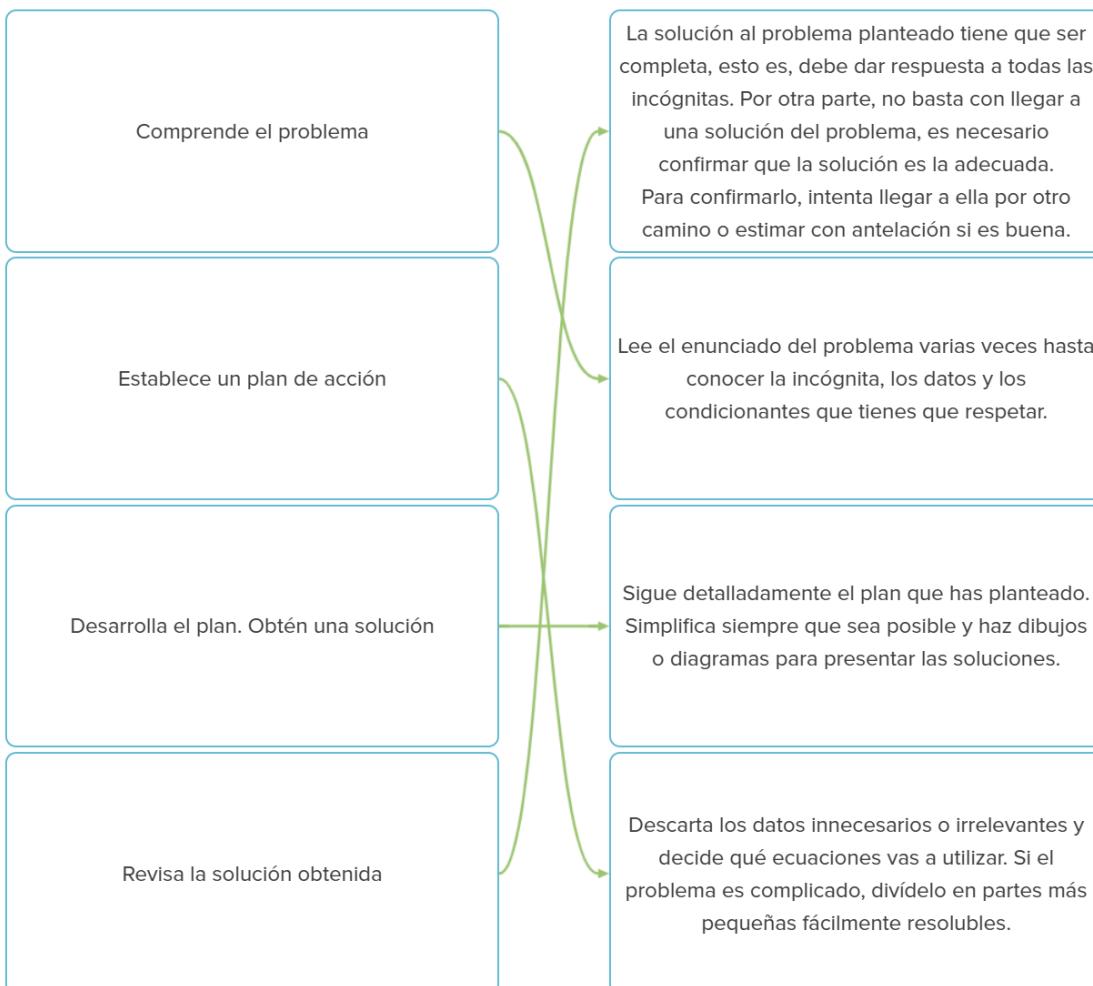
2. ● Ordena del 1 al 5 los siguientes pasos, que describen el pensamiento computacional:

1. Hacemos deducciones, planteamos hipótesis, imaginamos situaciones... Utilizamos el pensamiento abstracto. ✓
2. Simplificamos los elementos de un problema y lo dividimos en otros más sencillos. ✓
3. Identificamos el aspecto esencial de un problema (datos, condicionantes y restricciones). ✓
4. Desarrollamos un modelo que pueda ser una solución. ✓
5. La solución obtenida debe poder ser ejecutada por un sistema informático. ✓

3. ● Une cada palabra con su definición.



4. ● El método para solucionar problemas comprende cuatro fases. Une cada una de ellas con su explicación.



5. A lo largo de la unidad hemos aprendido que los datos son fundamentales en el pensamiento computacional. Indica a qué tipo de datos se refiere cada una de estas dos definiciones:

Alfanuméricos Numéricos

- a) Son representados mediante cifras. Pueden ser enteros (que permiten contar elementos) o decimales (que permiten expresar partes no completas de cantidades).

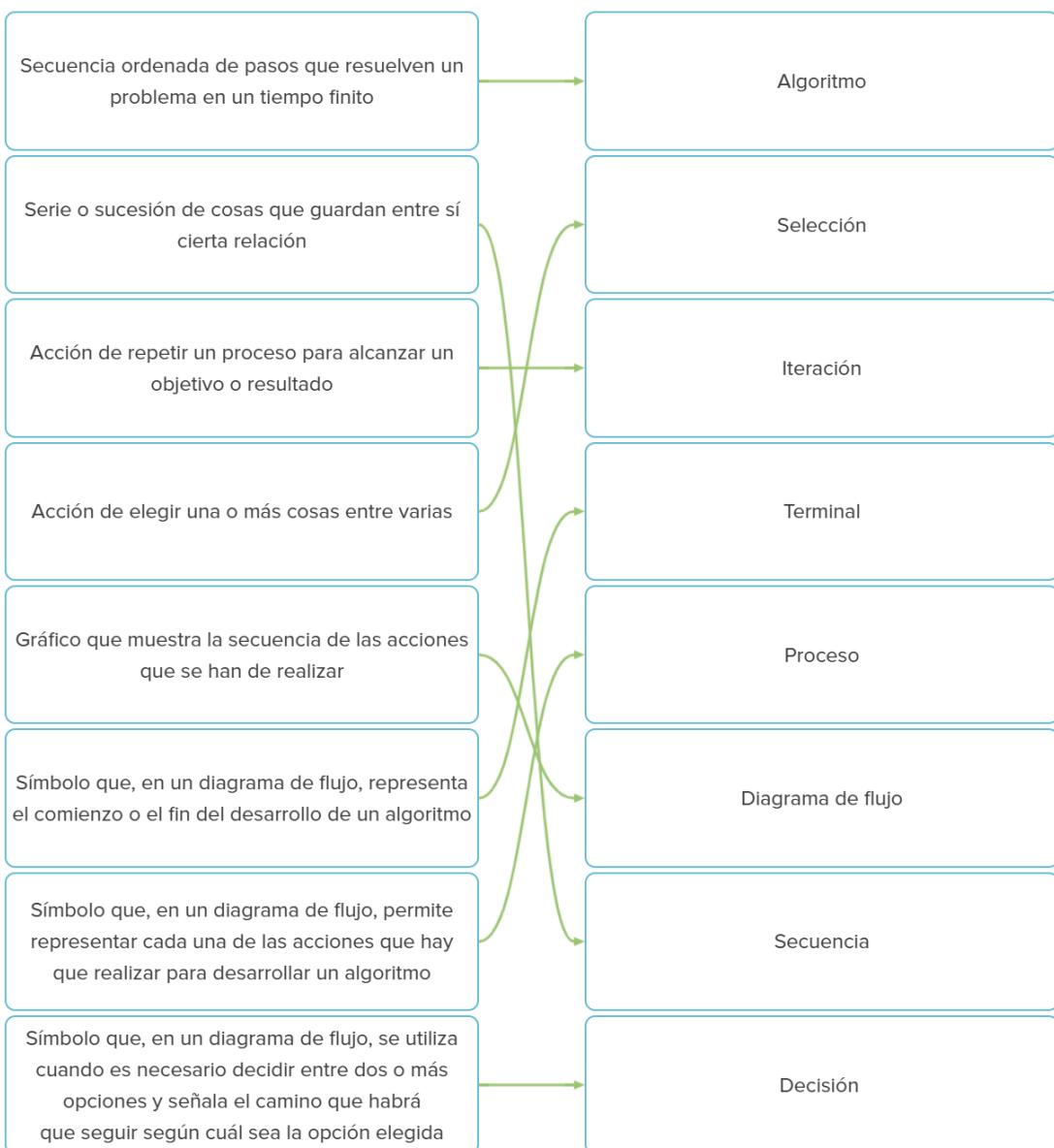
Numéricos

- b) Están formados por todo tipo de caracteres: cifras, letras, símbolos, etc. Ejemplos: las matrículas de los coches, las direcciones de correo electrónico y el número del NIF.

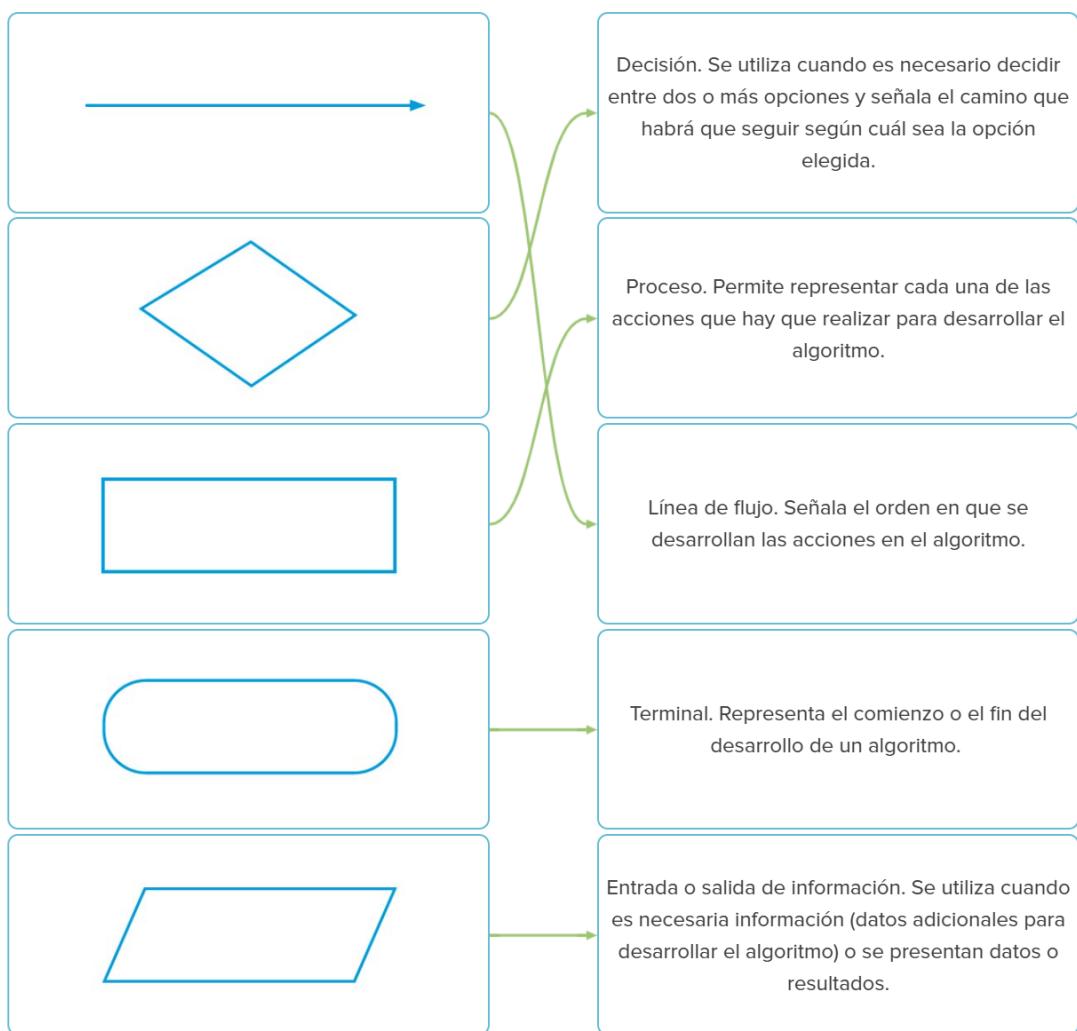
Alfanuméricos

Algoritmos

6. • • Une cada palabra con su definición.



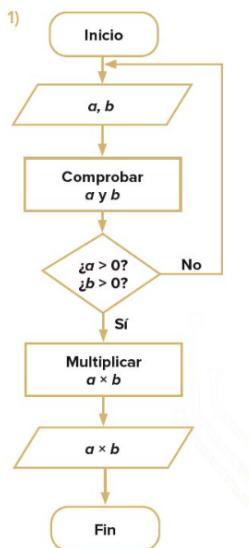
7. • Une cada símbolo de los diagramas de flujo con la función que representa.



8. • Di si las siguientes afirmaciones sobre tipos de algoritmos son verdaderas o falsas:

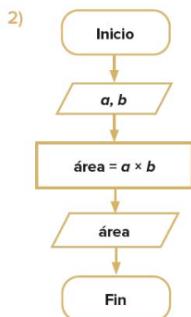
- a) En los algoritmos de estructura secuencial, las instrucciones se van cumpliendo en el orden en que aparecen.
 Verdadero Falso
- b) En los algoritmos de estructura secuencial, las instrucciones se llevan a cabo sólo si se cumple una determinada condición.
 Verdadero Falso
- c) Los algoritmos de estructura selectiva incluyen instrucciones que se repiten varias veces.
 Verdadero Falso

9. ● ● ¿Para qué nos sirve cada uno de los siguientes diagramas de flujo?



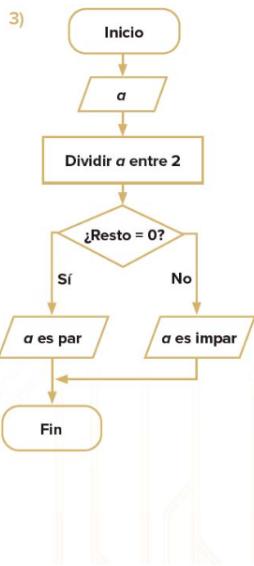
- a) Calcular el área de un rectángulo
- b) Calcular el perímetro de un rectángulo
- c) Multiplicar dos números
- d) Multiplicar dos números, siempre que ambos sean mayores que 0

- e) Saber si un número es par o impar
- f) Saber si un número es divisible entre dos o entre tres



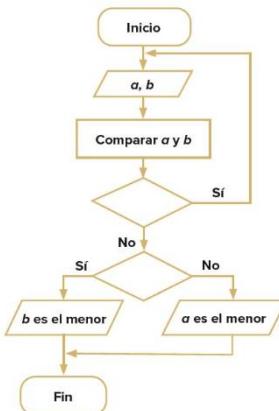
- 
- a) Calcular el área de un rectángulo
 - b) Calcular el perímetro de un rectángulo
 - c) Multiplicar dos números
 - d) Multiplicar dos números, siempre que ambos sean mayores que 0

 - e) Saber si un número es par o impar
 - f) Saber si un número es divisible entre dos o entre tres



- a) Calcular el área de un rectángulo
- b) Calcular el perímetro de un rectángulo
- c) Multiplicar dos números
- d) Multiplicar dos números, siempre que ambos sean mayores que 0
- e) Saber si un número es par o impar
- f) Saber si un número es divisible entre dos o entre tres

10. ● ● El siguiente diagrama de flujo incompleto sirve para comparar dos valores distintos y saber cuál es el menor de ellos. ¿Cuál es la pregunta en cada una de las dos decisiones que faltan?



a) La primera es $a > b$ y la segunda es $a < b$

b) La primera es $a = b$ y la segunda es $a > b$

c) Las dos preguntas son $a > b$

d) La primera es $a > b$ y la segunda es $a = b$

11. 🟠 🟡 Teniendo en cuenta las normas sobre cómo hacer diagramas de flujo, indica para cada imagen si es correcta o incorrecta.

a)



Correcta

Incorrecta

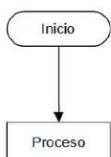
b)



Correcta

Incorrecta

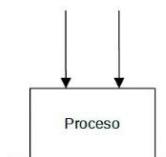
c)



Correcta

Incorrecta

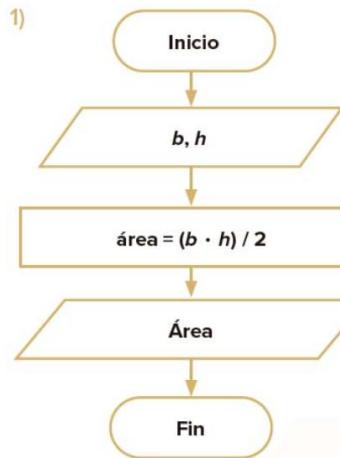
d)



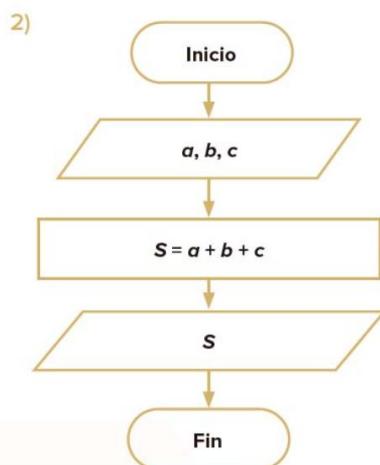
Correcta

Incorrecta

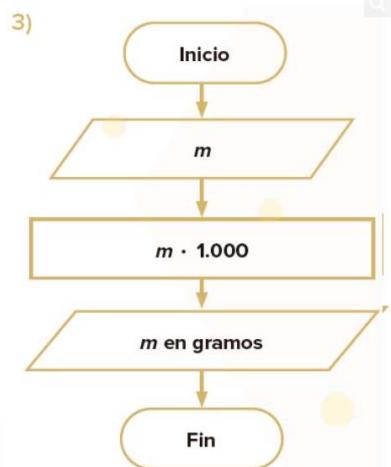
12. • • ¿Para qué nos sirve cada uno de los siguientes diagramas de flujo?



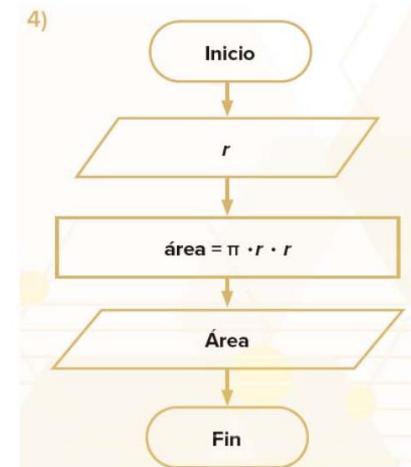
a) Calcular el área de un triángulo de base b y altura h ▾



c) Sumar tres números a, b y c ▾



b) Pasar una masa m de kilogramos a gramos ▾



d) Calcular el área de un círculo de radio r ▾

Programación

13.   Selecciona la respuesta correcta en cada caso:

1. Son instrucciones de la categoría **Control** en Scratch:

- a) detener todos y operaciones b) repetir y apariencia c) por siempre y movimiento d) por siempre y esperar hasta que
-

2. Son instrucciones de la categoría **Movimiento** en Scratch:

- a) girar 15 grados b) mover 10 pasos c) apuntar en dirección 90 d) Todas las respuestas anteriores son correctas.
-

3. En la categoría **Apariencia** encontramos bloques que nos permiten...

- a) Cambiar de un disfraz a otro. b) Apuntar en dirección 90. c) Crear variables. d) Ninguna de las respuestas anteriores es correcta.
-

4. En la categoría **Operadores** encontramos bloques que nos permiten...

- a) Realizar operaciones aritméticas. b) Unir textos. c) Realizar operaciones lógicas. d) Todas las respuestas anteriores son correctas.
-

14. De los siguientes programas, ¿cuál simulará mejor que un personaje ande?

a)



b)



c)



15. 🐞 Queremos que el escarabajo se mueva describiendo un triángulo. ¿Cuál es el bloque de instrucciones correcto?



a)

A Scratch script starting with a green flag button. It contains five blue "slide" blocks. The first block has x: 0 and y: 0. The subsequent four blocks each have a different path: (1, 100), (1, 100), (1, 0), and (1, 0).

```
al hacer clic en [bandera v-]
ir a x: (0) y: (0)
deslizar en (1) segs a x: (100) y: (0)
deslizar en (1) segs a x: (100) y: (100)
deslizar en (1) segs a x: (0) y: (100)
deslizar en (1) segs a x: (0) y: (0)
```

b)

A Scratch script starting with a green flag button. It contains four blue "slide" blocks. The first block has x: 0 and y: 0. The second block has a path of (1, 100). The third block has a path of (1, 50). The fourth block has x: 0 and y: 0.

✓

```
al hacer clic en [bandera v-]
ir a x: (0) y: (0)
deslizar en (1) segs a x: (100) y: (0)
deslizar en (1) segs a x: (50) y: (100)
deslizar en (1) segs a x: (0) y: (0)
```

c)

A Scratch script starting with a green flag button. It contains five blue blocks: a "move" block (100 steps), a "turn" block (120 degrees counter-clockwise), a "slide" block (1 segment to a random position), another "move" block (100 steps), and another "turn" block (120 degrees counter-clockwise).

```
al hacer clic en [bandera v-]
mover (100) pasos
girar (120) grados
deslizar en (1) segs a [posición aleatoria v-]
mover (100) pasos
girar (120) grados
```

16. ● ● ● Para el siguiente diagrama de bloques, señala si las afirmaciones son verdaderas o falsas.



- a) Hay una variable creada, llamada Número final, que va almacenando el valor de la suma.

Verdadero Falso

- b) El programa suma cuatro números que va escribiendo el jugador.

Verdadero Falso

- c) El programa finaliza mostrando en pantalla la suma de los números durante 2 segundos.

Verdadero Falso