Pilares del pensamiento computacional: Algoritmos

Algoritmos. Vamos paso a paso

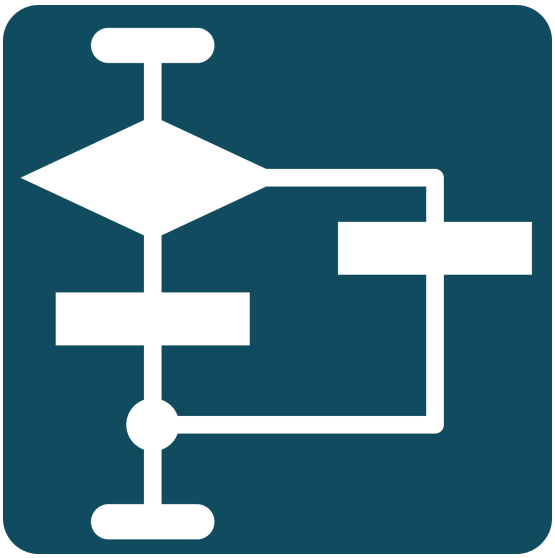
Copyright 2026



Un algoritmo es un plan, un **conjunto de instrucciones paso a paso** para resolver un problema. En un algoritmo se **identifica** cada instrucción y se **planifica** el orden en que se deben ejecutar.

¿Qué es un algoritmo?

Copyright 2026



Los algoritmos son uno de los cuatro pilares del pensamiento computacional. Un algoritmo es un **plan**, un conjunto de instrucciones paso a paso para resolver un problema. Si puedes atarte los cordones de los zapatos, preparar una taza de té, vestirte o preparar una comida, entonces ya sabes **cómo seguir un algoritmo**.

En un algoritmo se **identifica** cada instrucción y se **planifica** el orden en el que se deben llevar a cabo. Los algoritmos a menudo se usan como **punto de partida** para crear un programa de ordenador y, a veces, se escriben como un **diagrama de flujo** o en **pseudo-código**.

Si queremos decirle a un ordenador que haga algo, tenemos que escribir un programa informático que le diga al ordenador, paso a paso, exactamente **qué** queremos que haga y **cómo** queremos que lo haga. Este programa paso a paso necesitará planificación, y para hacerlo usamos un algoritmo.

Las ordenadores son tan buenos como los algoritmos que se les da. Si introduces en el ordenador un algoritmo deficiente, obtendrás un resultado deficiente.

Los algoritmos se utilizan para muchas cosas diferentes, incluidos los cálculos, el procesamiento de datos y la automatización.

Haciendo un plan

Es importante planificar la solución a un problema para asegurarse de que será correcta. Usando el pensamiento computacional y la descomposición, podemos dividir el problema en partes más pequeñas y luego podemos planificar cómo volver a encajarlo en un **orden adecuado** para resolver el problema.

Copyright 2026

Este orden se puede representar como un algoritmo. **Un algoritmo debe ser claro**.

Debe tener:

- un punto de partida
- un punto de llegada
- un conjunto de instrucciones claras entre ambos.



Representación de un algoritmo: Pseudo-código

Hay dos formas principales en que se pueden representar los algoritmos: **pseudo-código** y **diagramas de flujo**.

La mayoría de los programas se desarrollan utilizando **lenguajes de programación**. Estos lenguajes tienen una **sintaxis** específica que debe usarse para que el programa funcione correctamente. El **pseudo-código** no es un lenguaje de programación, es una forma simple de describir un conjunto de instrucciones que **no** tiene que usar una sintaxis específica.

Escribir en pseudo-código es **similar** a escribir en un lenguaje de programación. Cada paso del algoritmo está escrito en una **línea propia** en secuencia. Por lo general, las instrucciones se escriben en mayúsculas, las variables en minúsculas y los mensajes en forma de oración.

En pseudo-código, **ENTRADA** hace una pregunta. **SALIDA** imprime un mensaje en pantalla.

Se podría crear un programa simple para preguntarle a alguien su nombre y edad, y hacer un comentario basado en esto. Este programa representado en pseudo-código tendría este aspecto:

SALIDA “Cómo te llamas?”

ENTRADA el usuario introduce, teclea, su nombre

ALMACENAR la entrada del usuario en la variable **nombre**

SALIDA “Hola” + nombre

SALIDA “¿Qué edad tienes?”

ENTRADA el usuario introduce su edad

ALMACENAR la entrada del usuario en la variable **edad**

SI edad >= 70 ENTONCES

    SALIDA “¡Estás estupendo!”

SI NO

    SALIDA “¡Eres un chaval!”

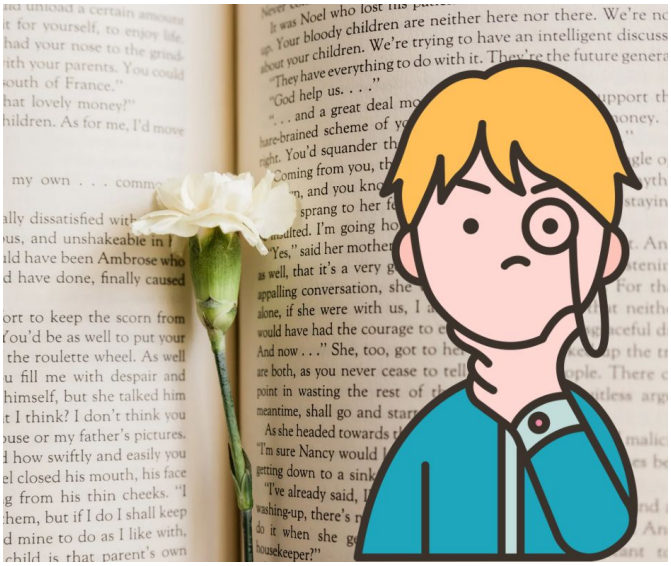
> **IDEA**. Pueden jugar a **guiar el dibujo** de otra persona. Una persona da las instrucciones, por ejemplo: dibuja un círculo, haz unos triángulos saliendo del círculo, dibuja una casa debajo del círculo…. Puede haber varias personas dibujando lo que una persona indica. Las

diferencias entre los dibujos en parte se deberán a la que **las instrucciones no han sido lo suficientemente concretas**. A medida que jueguen irán **afinando** las instrucciones, por ejemplo: dibuja un círculo de 3 cm de diámetro en el centro del la hoja, haz 5 triángulos iguales saliendo del círculo. Los triángulos tienen dos lados iguales, la base es el trazado del círculo y no es igual a los lados, dibuja una casa debajo del círculo, compuesta por un tejado triangular igual a los que has dibujado para hacer el sol…. Para facilitar el trabajo puedes ofrecerles papel cuadriculado para hacer esta actividad y entregarles fichas a los que deben describir para que las otras personas dibujen o que lo hagan de manera libre.

IDEA. Hacer un pseudo-código para hacer **tareas sencillas**, como ir al baño, sacar punta a un lápiz, levantar una silla….. Cada equipo escribe las instrucciones y se las pasa a otro equipo. El otro equipo las tiene que seguir al pie de la letra y hacer **solo** lo que pone en las instrucciones.

IDEA. Tal vez conozcas el texto de **Julio Cortazar** “Instrucciones para subir una escalera”. Puede ser divertido escoger textos de instrucciones y usarlos de base para hacer un pseudo-código. O coger un pseudo-código sencillo y convertirlo en un texto difícil de entender cómo es el caso de este cuento. También puedes adaptar el cuento para que no se sepa de qué se habla y jugar a adivinar. A partir de aquí puedes animarles ha hacer su propio cuento , en el que describan una actividad que las demás deban adivinar. Una posibilidad de **adaptación** podrías ser: “Instrucciones para realizar una tarea misteriosa”: "*Nadie habrá dejado de observar que con frecuencia el suelo se pliega de manera tal que una parte sube en ángulo recto con el plano del suelo, y luego la parte siguiente se coloca paralela a este plano, para dar paso a una nueva perpendicular, conducta que se repite en espiral o en línea quebrada hasta alturas sumamente variables. Agachándose y poniendo la mano izquierda en una de las partes verticales, y la derecha en la horizontal correspondiente, se está en posesión momentánea de un segmento. Cada uno de estos segmentos, formados como se ve por dos elementos, se sitúa un tanto más arriba y adelante que el anterior, principio que da sentido a la estructura, ya que cualquiera otra combinación producirá formas quizá más bellas o pintorescas, pero incapaces de cumplir su función. Estas estructuras se encaran de frente, pues hacia atrás o de costado resultan particularmente incómodas. La actitud natural consiste en mantenerse de pie, los brazos colgando sin esfuerzo, la cabeza erguida aunque no tanto que los ojos dejen de ver los peldaños inmediatamente superiores al que se pisa, y respirando lenta y regularmente. Para realizar nuestra tarea misteriosa se comienza por levantar esa parte del cuerpo situada a la derecha abajo, envuelta casi siempre en cuero o gamuza, y que salvo excepciones cabe exactamente en el segmento. Puesta en el primer segmento dicha parte, que para abreviar llamaremos pie, se recoge la parte equivalente de la izquierda (también llamada pie, pero que no ha de confundirse con el pie antes citado), y llevándola a la altura del pie, se le hace seguir hasta colocarla en el segundo segmento, con*

lo cual en éste descansará el pie, y en el primero descansará el pie. (Los primeros segmentos son siempre los más difíciles, hasta adquirir la coordinación necesaria. La coincidencia de nombre entre el pie y el pie hace difícil la explicación. Cuídese especialmente de no levantar al mismo tiempo el pie y el pie). Llegado en esta forma al segundo segmento, basta repetir alternadamente los movimientos hasta encontrarse con el final de la estructura. Se sale de ella fácilmente, con un ligero golpe de talón que la fija en su sitio, del que no se moverá hasta el momento del descenso"



IDEA. Jugar a ser **robots**: Se puede jugar por tríos: una persona es un robot tipo BeeBot, otra quien programa y la tercera al ver el código escrito tiene que **adivinar a dónde llegará** el “robot”. Quien programa escribe la secuencia usando flechas, se lo muestra a quien adivina y este hace una predicción. Luego se “programa” a quien hace de robot haciendo como que tiene una botonera en la espalda (adelante, atrás, izquierda, derecha). El juego aumenta la dificultad si quien hace de robot tiene los **ojos vendados** o si pones **obstáculos**. Facilita el juego hacerlo en un lugar donde haya baldosas o similares que delimiten cuál es la unidad de desplazamiento.

Copyright 2026



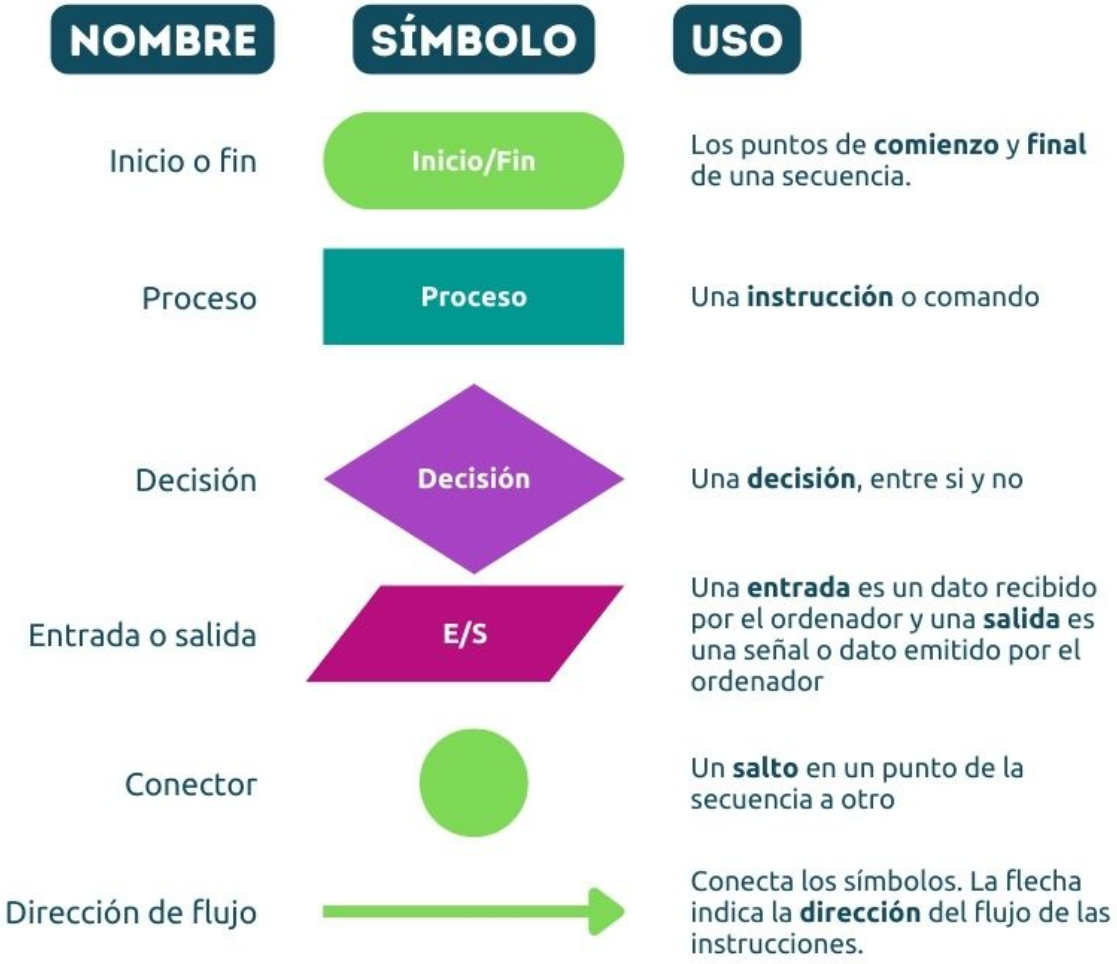
Si no conoces como funciona la Beebot te dejamos un video, podrás ver cómo funciona a partir de el minuto 1: 47

https://www.youtube.com/embed/sqqmH5PwR4s

Representación un algoritmo: diagramas de flujo

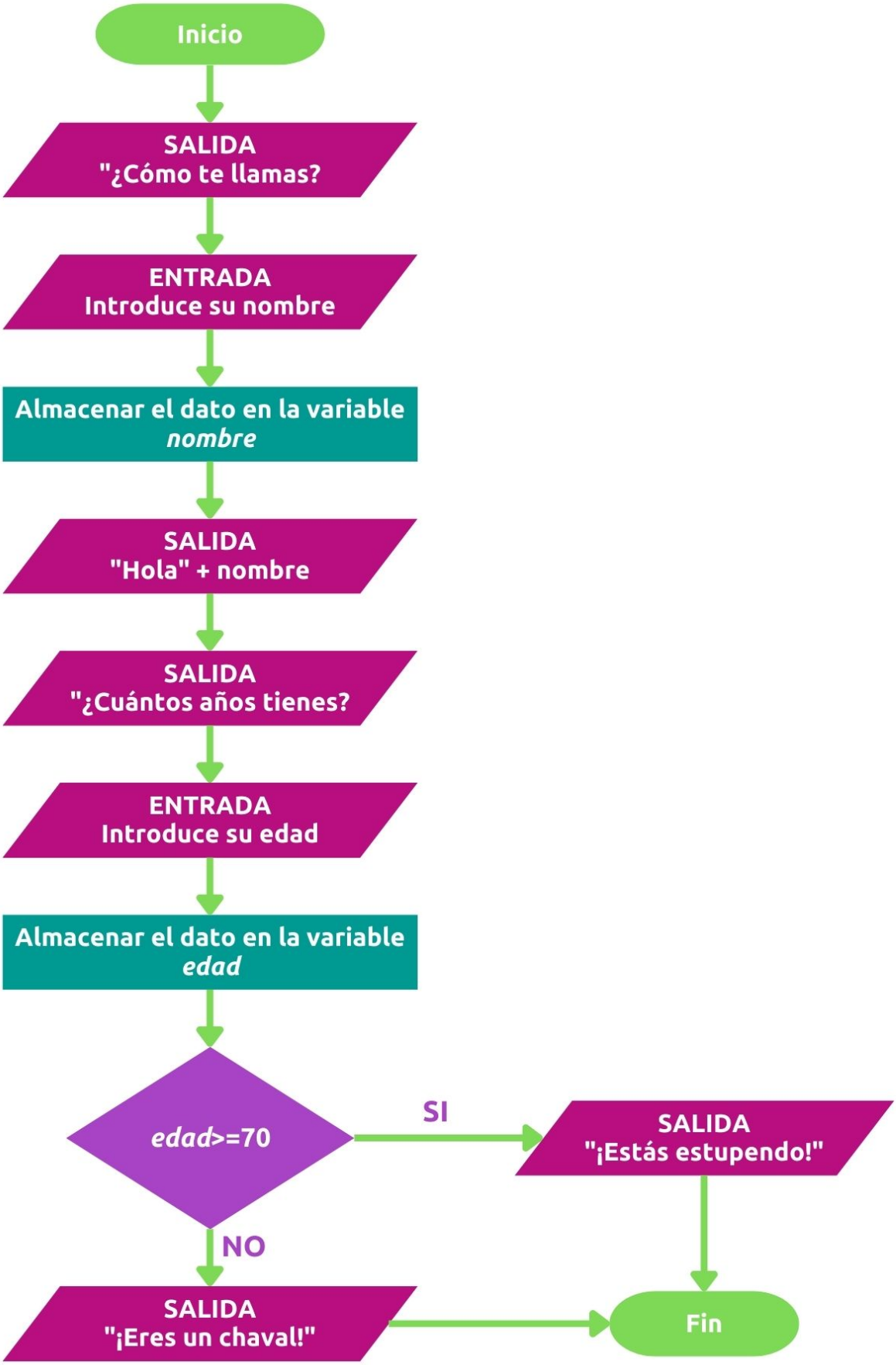
Un diagrama de flujo es una **representación** de un conjunto de instrucciones. Los diagramas de flujo normalmente usan **símbolos** estándar para representar las diferentes instrucciones. Hay pocas reglas reales sobre el **nivel de detalle** necesario en un diagrama de flujo. A veces, los diagramas de flujo se dividen en muchos pasos para brindar muchos detalles sobre lo que está sucediendo exactamente. A veces se simplifican para que una serie de pasos ocurran en un solo paso.

Símbolos del diagrama de flujo



Se podría crear un programa simple para preguntarle a alguien su nombre y edad, y hacer un comentario basado en esto. Este programa **representado** como un diagrama de flujo se vería así:

Copyright 2026



Los diagramas de flujo pueden emplearse para resolver problemas de cualquier naturaleza, sin

importa el tema ni la dificultad (de hecho, cuanto más complejo es el problema más útil) Veamos un **ejemplo**:

https://www.youtube.com/embed/3sU6KIxjqMI

> **IDEA**. Se pueden crear diagramas de flujo con lápiz y papel pero también hay aplicaciones interesante para hacer diagramas de flujo. Una de ellas es **Diagrams**. Esta aplicación gratuita y de código abierto ayuda a crear una gran variedad de diagramas en Windows, Linux y macOS, agregando imágenes, configurando los textos y eligiendo el tipo de fuente, etc. El alumnado tendrá la opción de diseñar diagramas de flujo, mapas mentales.... y otro tipo de representaciones aunque para este curso con las que hemos nombrado tenemos suficiente. Su interfaz es muy práctica e intuitiva e incorpora capas que facilitan la edición. Otra opción es **Mindomo**. En este caso no hablamos de software libre. Se trata de un software colaborativo en línea para realizar diagramas de flujo, mapas mentales... y otras muchas cosas... aunque para este curso estas nos parecen las más adecuadas. Software Freemium ofrece sus servicios básicos de forma gratuita, si bien se cobra por las funciones prémium. La versión gratuita tiene un límite de tres mapas mentales y funciones prémium inhabilitadas, como carga de audio/video, descarga de PDF, PowerPoint y Excel y protección con contraseña. El software se puede usar desde cualquier navegador web estándar o instalando la aplicación gratuita para escritorio, iPad y Android.

Revision #2
Created 2023-06-22 13:52:46 CEST by Ana López Floría
Updated 2023-06-22 13:55:55 CEST by Ana López Floría

Copyright 2026