

A WAKE LAB

#programmingbootcamp



FUNDAMENTOS DE PROGRAMACIÓN

The image is a collage of screenshots from the game DOOM. It features several levels including Hell, Helltrain, and Helltrain Station. Overlaid on the images are two main text elements: 'AWAKELAB #programmingbootcamp' in large white letters at the top left, and 'FUNDAMENTOS DE PROGRAMACIÓN' in large white letters in the center. The background consists of various DOOM levels with visible textures and enemies.

FUNDAMENTOS DE PROGRAMACIÓN



¿QUÉ VAMOS A VER?

- Introducción a los algoritmos.
- Definición de un algoritmo.
- Partes de un algoritmo.
- Análisis de casos.
- Variables y tipos de dato.
- Expresiones aritméticas y operadores.
- Estructuras de control condicional.



INTRODUCCIÓN A LOS ALGORITMOS



¿QUÉ ES UN ALGORITMO?



¿QUÉ ES LA LÓGICA ALGORÍTMICA?



DEFINICIÓN DE UN ALGORITMO

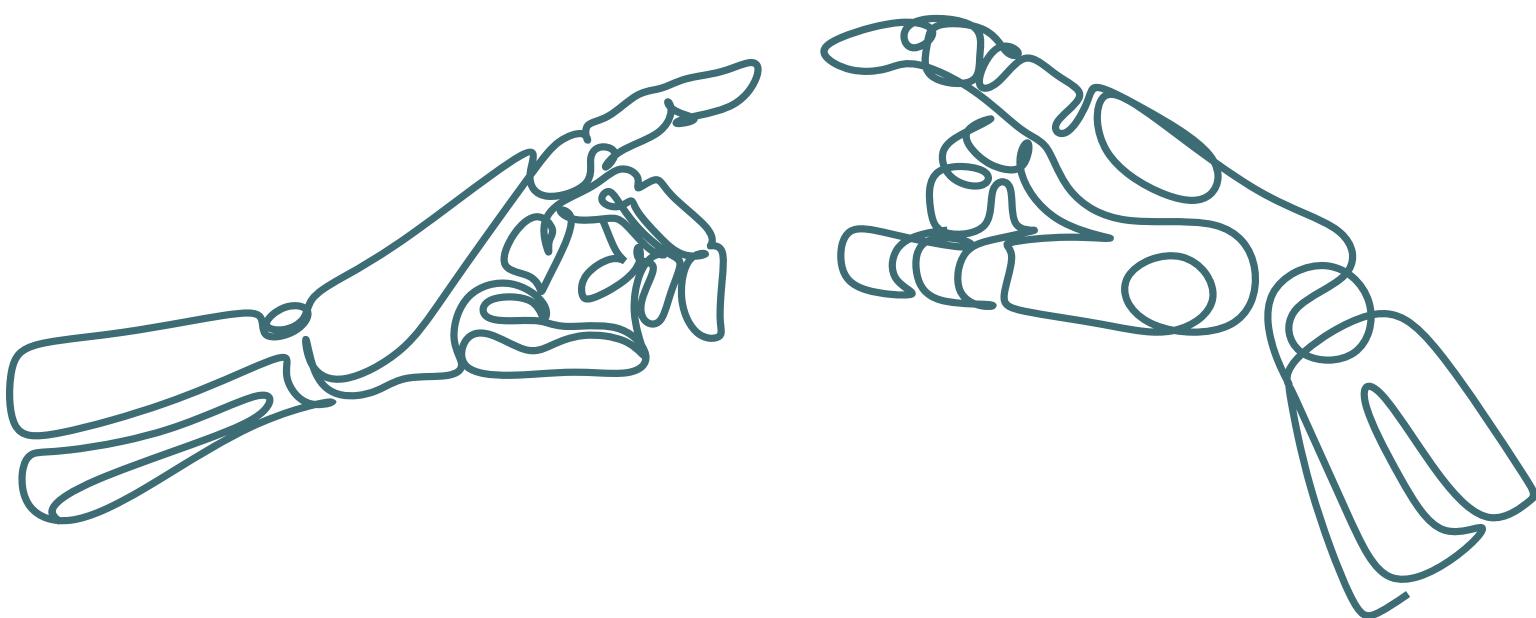
- Conjunto de instrucciones definidas.
- Permiten solucionar un problema.
- Siguen pasos sucesivos.
- Entregan un dato o realizan una acción final.
- Son independientes de cualquier lenguaje.
- Un algoritmo siempre debe ser **finito, bien definido y efectivo.**





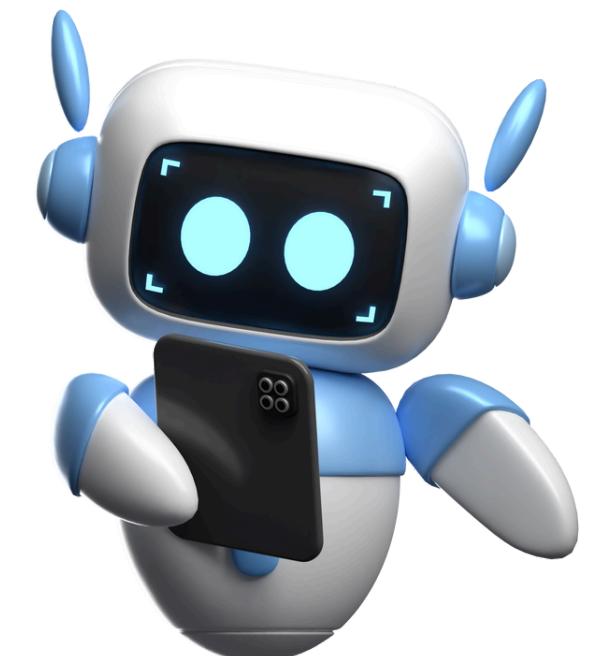
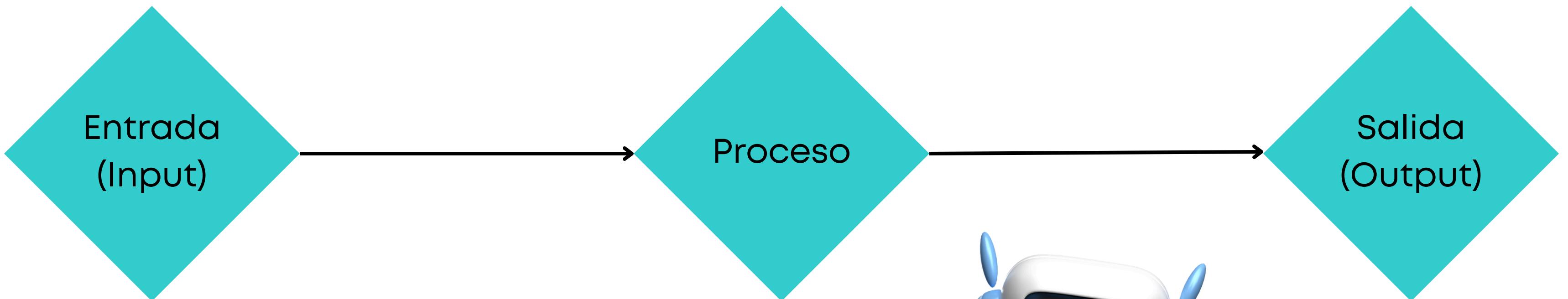
¿CÓMO DISEÑAR UN ALGORITMO?

- Primero que todo, entender que un algoritmo siempre va a tener al menos una **entrada** y una **salida**.
- Entender la problemática que se quiere solucionar.
- Analizar la mejor manera de abordar el problema.
- Ejecutar el plan de acción.
- Poner a prueba el algoritmo construido y reflexionar sobre su aplicación.





PARTES DE UN ALGORITMO





PARTES DE UN ALGORITMO

- Un algoritmo tiene una gama amplia de soluciones.
- Siempre es bueno evaluar el algoritmo creado.
- Analiza su diseño, tiempo de ejecución y uso de recursos.
- Si es necesario, optimiza el algoritmo.



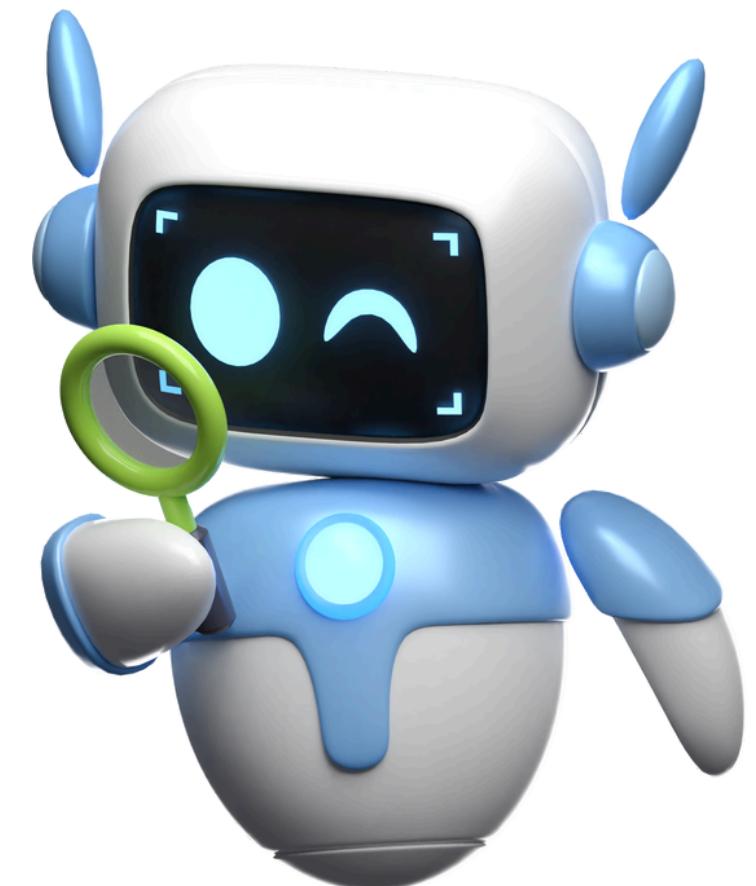


ANÁLISIS DE CASOS

Ejercicio: Calcular el promedio de dos números.

Solución:

1. *INICIO*
2. Ingresar los dos números (a y b).
3. Sumar los dos números ($suma = a + b$).
4. Dividir el resultado de la suma por 2 ($promedio = suma / 2$).
5. Mostrar el resultado del promedio de los números.
6. *FIN*



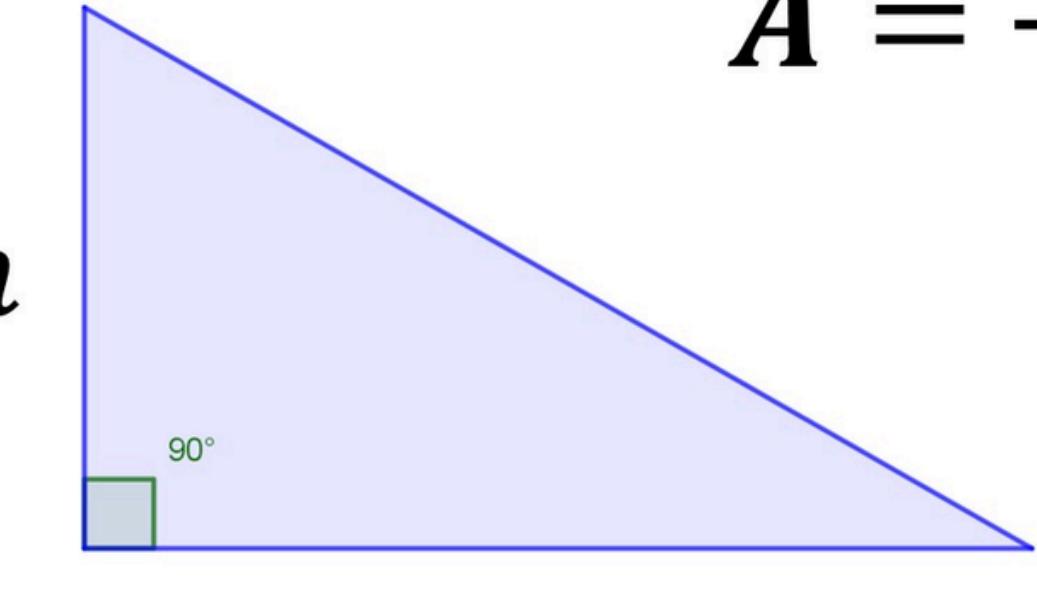


ANÁLISIS DE CASOS

Ejercicio: Calcular el área de un triángulo rectángulo.

Solución:

1. *INICIO*
2. Identificar las medidas de la base y la altura.
3. Multiplicar la base por la altura y dividir por 2 el resultado.
4. Mostrar el resultado.
5. *FIN*

$$A = \frac{bh}{2}$$




ANÁLISIS DE CASOS

Ejercicio: Averiguar si un número es primo o no.

Solución:

1. *INICIO*
2. Poner X igual a 2 ($X = 2$, X, variable que representa a los posibles divisores de N).
3. Dividir N por X (N/X).
4. Si el resultado es entero, entonces N no es primo y saltas al punto 9 (En caso contrario, continua el proceso).
5. Incrementar X en una unidad.
6. Si X es menor que N saltar al punto 3 (En caso contrario, continua el proceso en el punto 7).
7. Declarar N es primo,
8. Saltar al *FIN* (punto 10).
9. Declarar N no es primo.
10. *FIN*

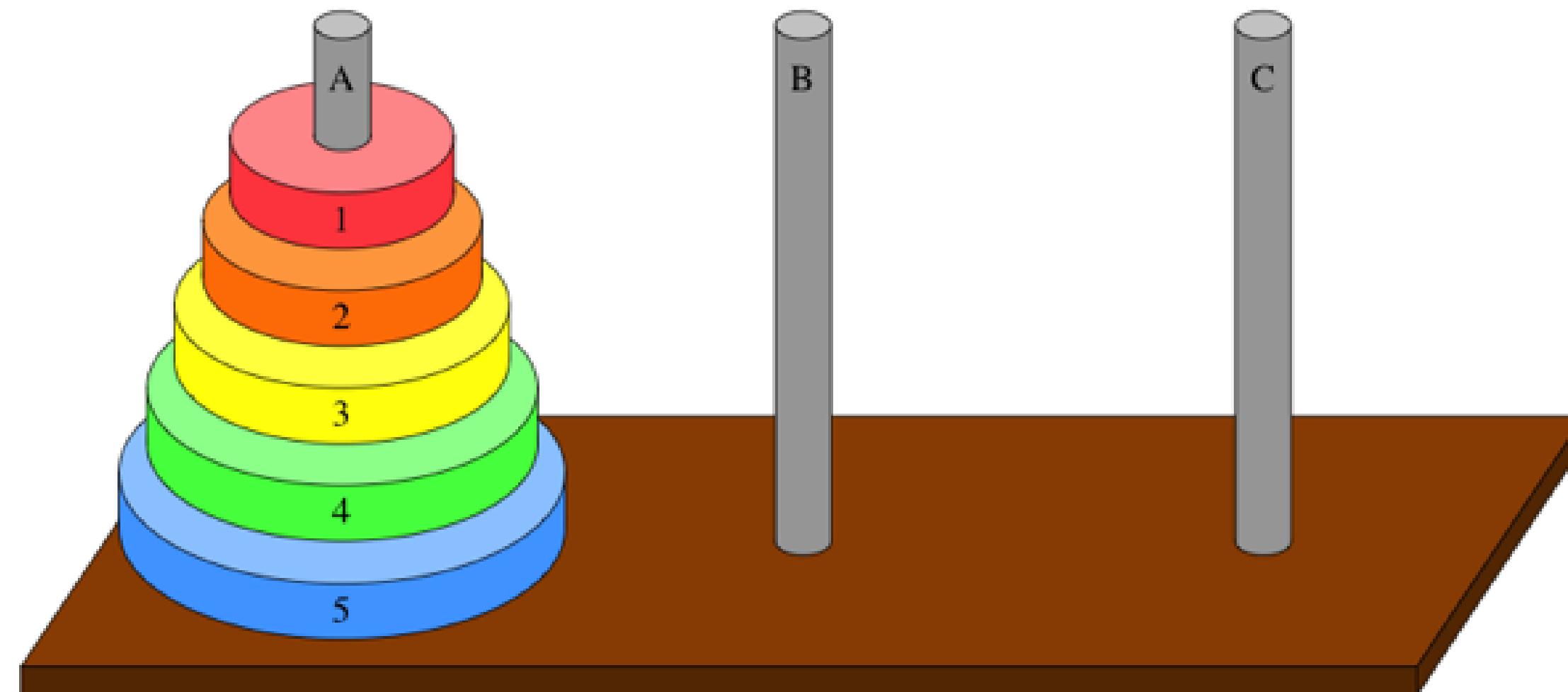
	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100



ANÁLISIS DE CASOS

Ejercicio: Las torres de Hanoi.

Pasar los bloques (uno a uno) a la columna siguiente o anterior de modo que lleguen en el mismo orden a la columna C.

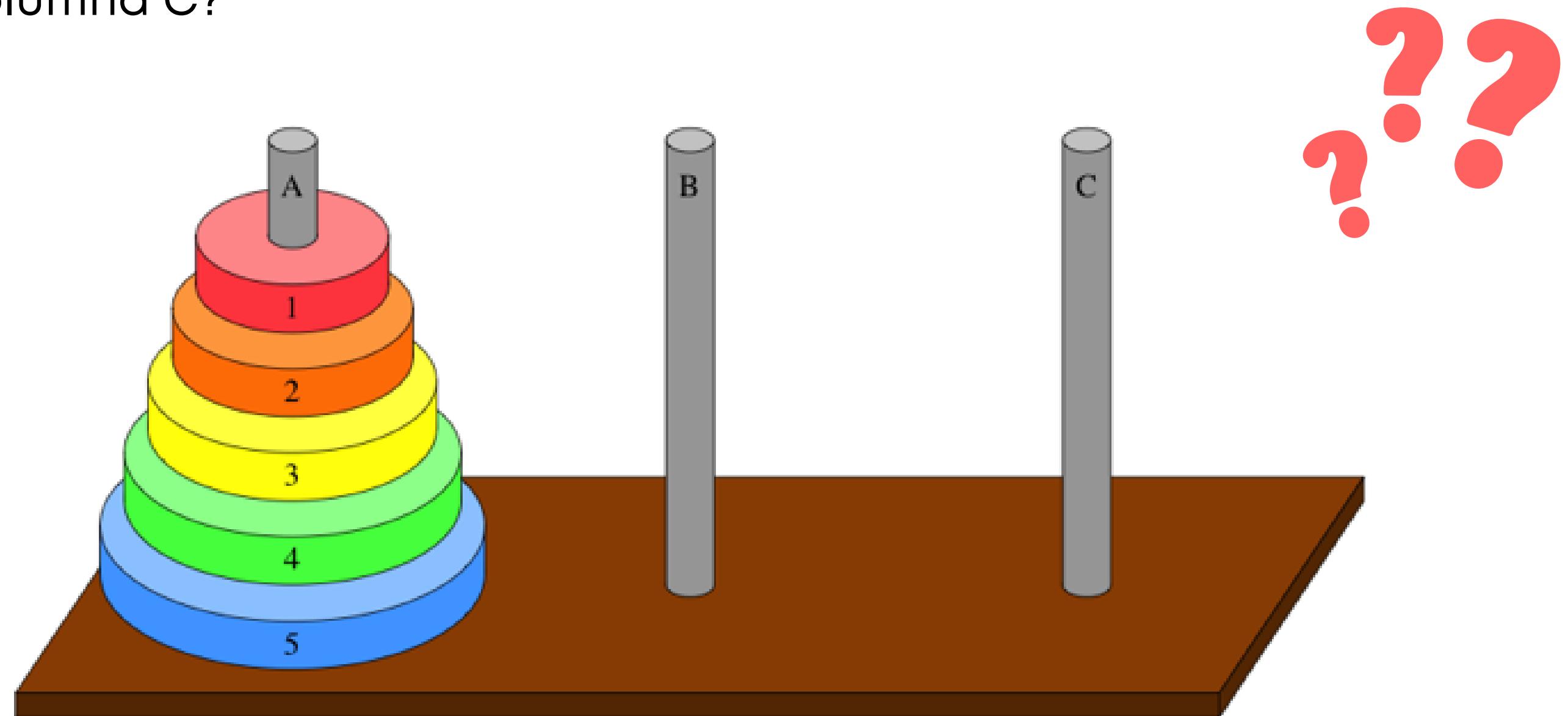




ANÁLISIS DE CASOS

Ejercicio: Las torres de Hanoi.

¿Cómo cambiaría nuestro algoritmo si ahora necesitamos que los bloques lleguen invertidos a la columna C?



FUNDAMENTOS DE PROGRAMACIÓN



<< UN ALGORITMO DEBE VERSE PARA CREERSE >> --DONALD KNUTH--



AWAKELAB

#programmingbootcamp

nodovirtual.awakelab.cl

 jELOU futurejob by  Adalid Chile