

Introducción a la Programación: Algoritmos Básicos

En esta presentación, exploraremos los fundamentos de la programación a través de algoritmos básicos. Estos algoritmos son la base para resolver problemas computacionales y automatizar tareas. Comenzaremos con ejemplos simples y progresivamente aumentaremos la complejidad, utilizando pseudocódigo y diagramas de flujo para visualizar cada paso del proceso.

Cálculo del Tiempo Medio por Kilómetro

El algoritmo que presentamos calcula el tiempo medio que un ciclista necesita para recorrer un kilómetro. El algoritmo recibe como entrada la distancia total recorrida (15 kilómetros) y el tiempo total empleado (40 minutos). Divide el tiempo total entre la distancia total para obtener el tiempo medio por kilómetro.

Entrada

Distancia total (15 km)

Tiempo total (40 minutos)

Salida

Tiempo medio por kilómetro (minutos)

Proceso

Dividir el tiempo total entre la distancia total.



Pseudocódigo para el Cálculo del Tiempo Medio

El pseudocódigo es una forma de expresar un algoritmo en lenguaje natural, utilizando una estructura similar a un lenguaje de programación. Aquí se presenta el pseudocódigo para calcular el tiempo medio por kilómetro:

```
Inicio
  distancia <- 15
  tiempo <- 40
  tiempo_medio <- tiempo / distancia
  Mostrar "El tiempo medio por kilómetro es: ", tiempo_medio, " minutos"
Fin
```

Diagrama de Flujo para el Cálculo del Tiempo Medio

El diagrama de flujo es una representación visual del algoritmo, utilizando símbolos y flechas para indicar las operaciones y el flujo de datos. Este diagrama de flujo ilustra el proceso de cálculo del tiempo medio por kilómetro:



Conversión de Grados Centígrados a Kelvin

Este algoritmo convierte una temperatura dada en grados Centígrados a la escala Kelvin. La fórmula de conversión es $K = C + 273.15$, donde K es la temperatura en Kelvin y C es la temperatura en grados Centígrados.

1

Entrada

Temperatura en grados Centígrados (C)

2

Salida

Temperatura en Kelvin (K)

3

Proceso

Aplicar la fórmula $K = C + 273.15$ para convertir la temperatura de grados Centígrados a Kelvin.



Pseudocódigo para la Conversión a Kelvin

Este pseudocódigo muestra cómo convertir una temperatura en grados Centígrados a Kelvin, utilizando la fórmula $K = C + 273.15$.

Inicio

Leer temperaturaC

temperaturaK \leftarrow temperaturaC + 273.15

Mostrar "La temperatura en Kelvin es: ", temperaturaK

Fin



Diagrama de Flujo para la Conversión a Kelvin

El diagrama de flujo para la conversión de grados Centígrados a Kelvin se representa a continuación. Este diagrama visualiza las diferentes etapas del proceso de conversión.





Cálculo de la Nota Final de un Estudiante

El algoritmo calcula la nota final de un estudiante en una materia, considerando las notas de dos prácticas y un examen final. Las prácticas representan el 40% de la nota final, mientras que el examen final representa el 60%.

Entrada	Descripción
practica1	Nota de la primera práctica
practica2	Nota de la segunda práctica
examen	Nota del examen final

El algoritmo calcula el promedio de las prácticas y lo multiplica por 0.4; multiplica la nota del examen por 0.6. Finalmente, suma ambos resultados para obtener la nota final.

Pseudocódigo para el Cálculo de la Nota Final

El pseudocódigo que se presenta a continuación calcula la nota final de un estudiante, tomando en cuenta las notas de las prácticas y el examen final, y aplicando los porcentajes de ponderación correspondientes.

Inicio

Leer practica1, practica2, examen

$\text{promedio_practicas} \leftarrow (\text{practica1} + \text{practica2}) / 2$

$\text{nota_final} \leftarrow (\text{promedio_practicas} * 0.4) + (\text{examen} * 0.6)$

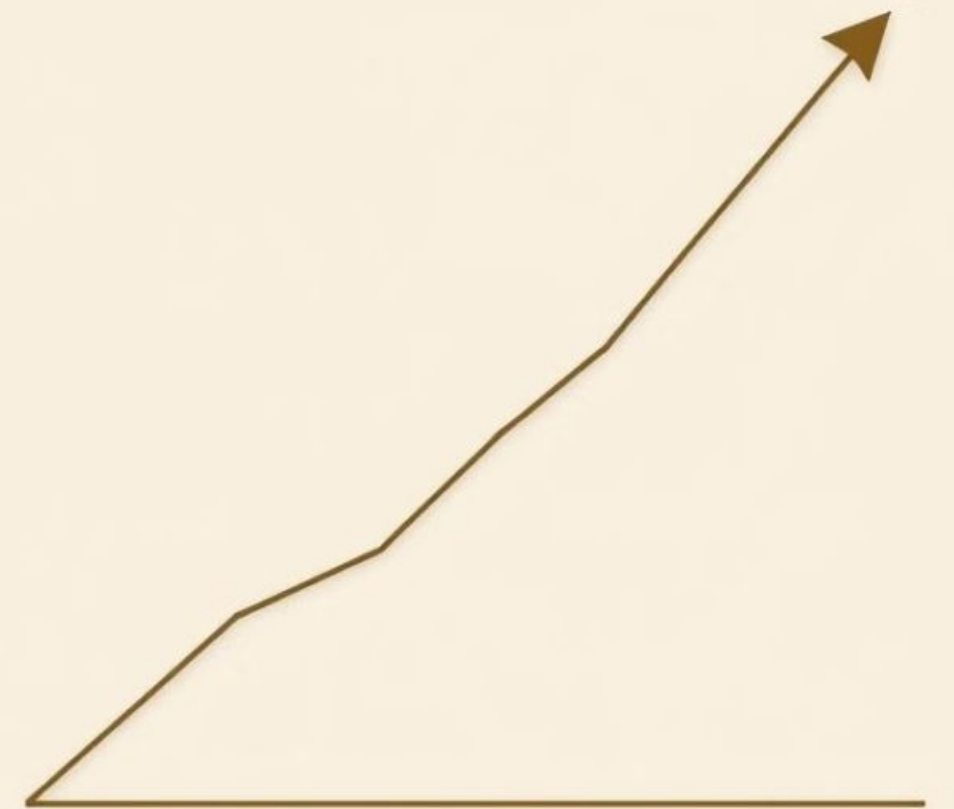
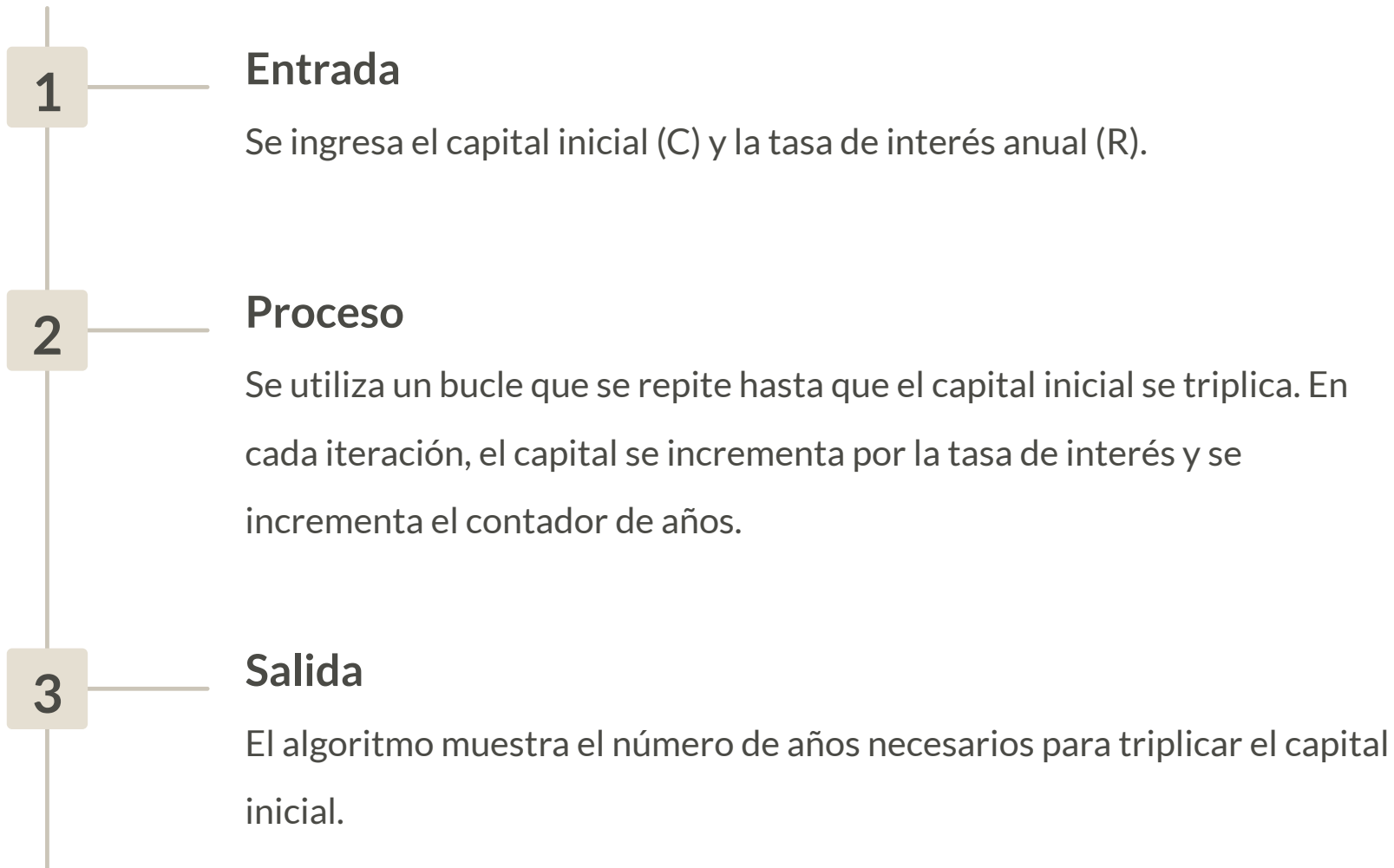
Mostrar "La nota final es: ", nota_final

Fin



Cálculo del Tiempo para Triplicar un Capital

Este algoritmo calcula el número de años necesarios para triplicar un capital inicial, considerando una tasa de interés anual fija.



1. Reading to become,
5: Do not the e dr oe have tuem cotheurtcly
2. Clase is be g a have sode scaneg,
have lunkt bee isc. —————

Pseudocódigo para el Cálculo del Tiempo

El pseudocódigo es una forma de representar un algoritmo utilizando un lenguaje simple y comprensible.

```
Inicio
  Leer capital, tasa
  años <- 0
  capital_objetivo <- capital * 3
  Mientras capital < capital_objetivo Hacer
    capital <- capital + (capital * tasa / 100)
    años <- años + 1
  Fin Mientras
  Mostrar "Se necesitan ", años, " años para triplicar el capital."
Fin
```

Mostrar Números Mayores o Iguales a 10

Este algoritmo lee 15 números y muestra solo aquellos que son mayores o iguales a 10.

1

Lectura de Números

Se lee un número en cada iteración del bucle.

2

Comparación

Se compara el número leído con el valor 10. Si el número es mayor o igual a 10, se muestra.



Pseudocódigo para Mostrar Números

Este pseudocódigo representa el algoritmo que lee 15 números y muestra aquellos que son mayores o iguales a 10.

Inicio

Para i desde 1 hasta 15 Hacer

Leer numero

Si numero ≥ 10 Entonces

Mostrar numero

Fin Si

Fin Para

Fin



Conversión de Dólares a Euros

Este algoritmo suma los precios de 7 artículos en dólares y luego los convierte a euros utilizando una tasa de cambio.

1

Entrada

Se ingresa la tasa de cambio y los precios de 7 artículos en dólares.

2

Suma

Se suma el precio de cada artículo.

3


Conversión

Se multiplica el total en dólares por la tasa de cambio para obtener el total en euros.

4

Salida

Se muestra el total en euros.



A wāckime

Pseudocódigo para Conversión de Moneda

Este pseudocódigo representa el algoritmo que suma los precios de 7 artículos en dólares y los convierte a euros.

```
Inicio
    total <- 0
    Leer tasa_cambio
    Para i desde 1 hasta 7 Hacer
        Leer precio
        total <- total + precio
    Fin Para
    total_euros <- total * tasa_cambio
    Mostrar "Total en euros: ", total_euros
Fin
```

Algoritmo para registrar el consumo en un café

Este algoritmo se encarga de calcular el pago de cada cliente en un café, aplicando un descuento del 15% si el consumo supera los 30,000.

El algoritmo también calcula el total de todos los pagos, brindando una visión general de las ganancias del día.

1 Entrada

Consumo de cada cliente y número de clientes.

2 Salida

Pago total de cada cliente (aplicando descuento si es necesario) y total de todos los pagos.

3 Proceso

Leer el consumo de cada cliente. Si el consumo es mayor a 30,000, aplicar un descuento del 15%. Calcular el total de pagos de todos los clientes.



Pseudocódigo para el algoritmo de consumo en un café

El pseudocódigo es una representación textual del algoritmo, utilizando un lenguaje sencillo y comprensible.

Este pseudocódigo describe paso a paso cómo funciona el algoritmo, incluyendo la entrada, el proceso y la salida.

```
Inicio
  Leer num_clientes
  total_general <- 0
  Para i desde 1 hasta num_clientes Hacer
    Leer consumo
    Si consumo > 30000 Entonces
      pago <- consumo * 0.85
    Sino
      pago <- consumo
    Fin Si
    total_general <- total_general + pago
    Mostrar "El pago del cliente ", i, " es: ", pago
  Fin Para
  Mostrar "El total de todos los pagos es: ", total_general
Fin
```



Diagrama de flujo para el algoritmo de consumo en un café

El diagrama de flujo es una representación visual del algoritmo, utilizando símbolos estándar para representar las diferentes operaciones.

Este diagrama ayuda a comprender el flujo de información y las decisiones que se toman en el algoritmo.





Algoritmo para calcular la hora anterior

Este algoritmo recibe la hora actual (hora, minutos y segundos) y calcula la hora en el segundo anterior, ajustando los valores para que sean válidos en el formato de reloj de 24 horas.

El algoritmo tiene en cuenta los casos especiales donde los segundos o los minutos son 0, asegurando que los valores de la hora se ajusten correctamente.

1

Entrada

Hora (H), Minutos (M), Segundos (S).

2

Salida

Hora, minutos y segundos ajustados para el segundo anterior.

3

Proceso

Si los segundos son 0, ajustar los minutos y posiblemente la hora. Si los minutos son 0 y los segundos también, ajustar la hora.

Pseudocódigo para calcular la hora anterior

El pseudocódigo para este algoritmo muestra cómo calcular la hora en el segundo anterior, incluyendo los ajustes necesarios para que los valores sean válidos en el formato de reloj de 24 horas.

Inicio

Leer H, M, S

Si S = 0 Entonces

S ← 59

Si M = 0 Entonces

M ← 59

Si H = 0 Entonces

H ← 23

Sino

H ← H - 1

Fin Si

Sino

M ← M - 1

Fin Si

Sino

S ← S - 1

Fin Si

Mostrar "La hora en el segundo anterior es: ", H, ":", M, ":", S

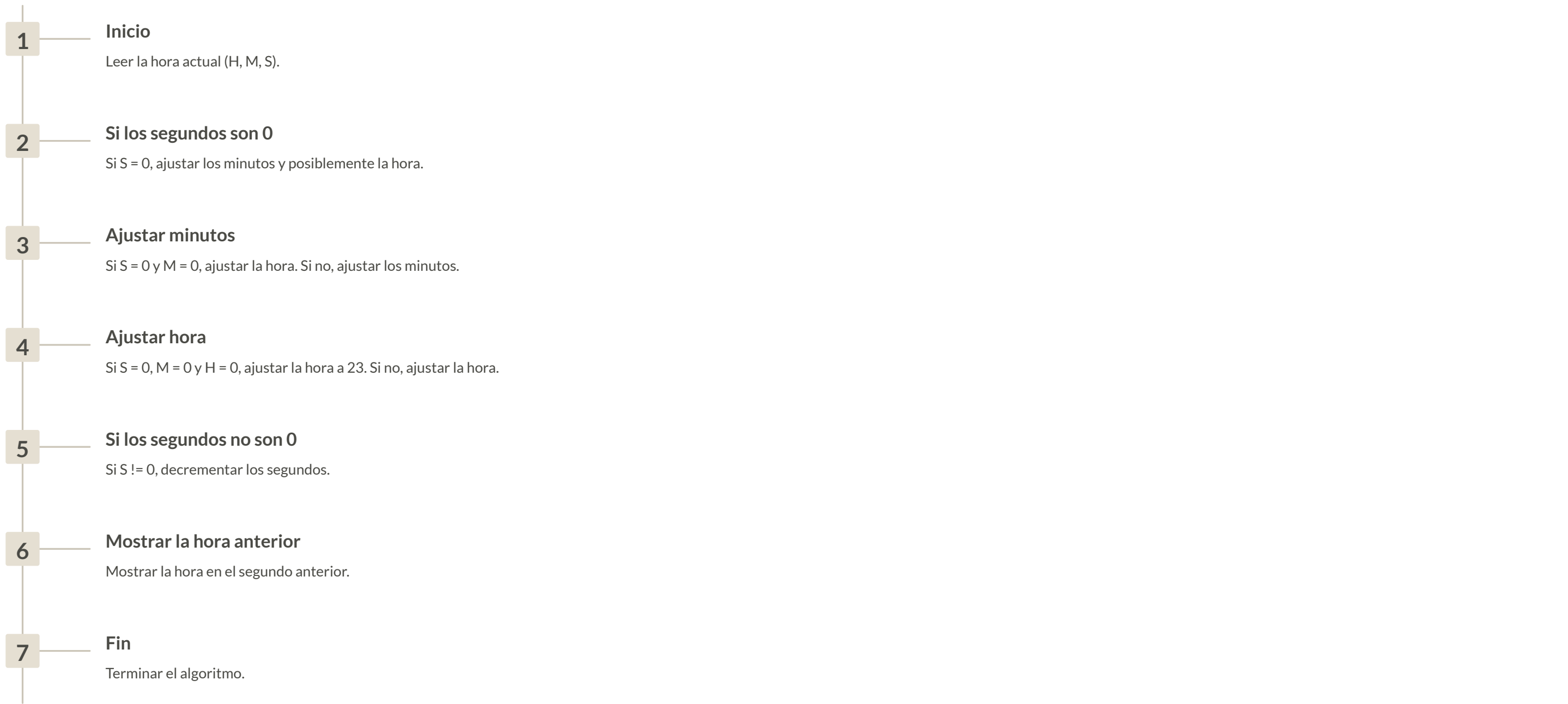
Fin

```
4 soreiccode Siliin(l)
5 reperiongaice
6
7
4 agilation for deckespours):
15 sotakinr fo(!- fercolles propriatong;
15 Dower = inpreiation)
21 Jeces
10 trealanot eildy essayne();
19 teuesridf: interristall, := Viollon();
19 27 Jeece:
17 Thaklcapet (lnes / - 99/4);
36 cortrangsorating!
19 sroume incieciatons!)
36
43
46 diingzähle: Knde (nce pecortecalsgacheff:
37 trop (f - tesise)
18 trcällie for erchione(Io) awe stlte fiese
18 thop := ~fercetiing)
23 stromtixe for cetiomilie weed pecetatial::
44 fiome if lessicowate inetastiaf)
10 tinee := false.
27 finur := toy (hangpest crde::
21 forcalic libowviess :
15 percamtion aid: {
10 a cortigutrethatte sote pepolitiatory
10 idatryiiaions weed aotergaeleale:
17 to lvelsag:
15 forcatile for heed ohte tesäly int hespirt.
79
46 fiuse / joat (sibous)
-caffin: (uber: cortister"7)
10 ioue: / howd110(- 8047));
19 reliött: ShedfietationsK);
27 imiöt / io perlettuae (f = sorticlle (Impactileg Inguities)
48
15 tanngelence. := ssionne),
gate: f: (ylbowll" crrich(r))
12 fines / estrisiotons)
36 > play:
34 #0. → / ineatine)
30 aletrifiall testatedy: (reen)
27 recme := tave lessasing cileck(t)
16
22
15
12
17
19
17
```


Diagrama de flujo para calcular la hora anterior

El diagrama de flujo visualiza el algoritmo para calcular la hora en el segundo anterior, mostrando el flujo de decisiones y acciones.

Este diagrama facilita la comprensión de cómo funciona el algoritmo, especialmente para aquellos que no están familiarizados con el pseudocódigo.





Algoritmo para calcular la suma de números

Este algoritmo recibe un número N y calcula la suma de todos los números desde 1 hasta N.

El algoritmo utiliza una estructura de repetición para iterar a través de los números y acumular la suma.

1

Entrada

Número N.

2

Salida

Suma de los números desde 1 hasta N.

3

Proceso

Utilizar una estructura de repetición para sumar los números desde 1 hasta N.

Pseudocódigo para calcular la suma de números

El pseudocódigo para este algoritmo muestra cómo calcular la suma de los números desde 1 hasta N utilizando una estructura de repetición.

Este pseudocódigo es un ejemplo de cómo se puede expresar un algoritmo de forma clara y concisa.

Inicio

Leer N

suma \leftarrow 0

Para i desde 1 hasta N Hacer

 suma \leftarrow suma + i

Fin Para

Mostrar "La suma desde 1 hasta ", N, " es: ", suma

Fin

