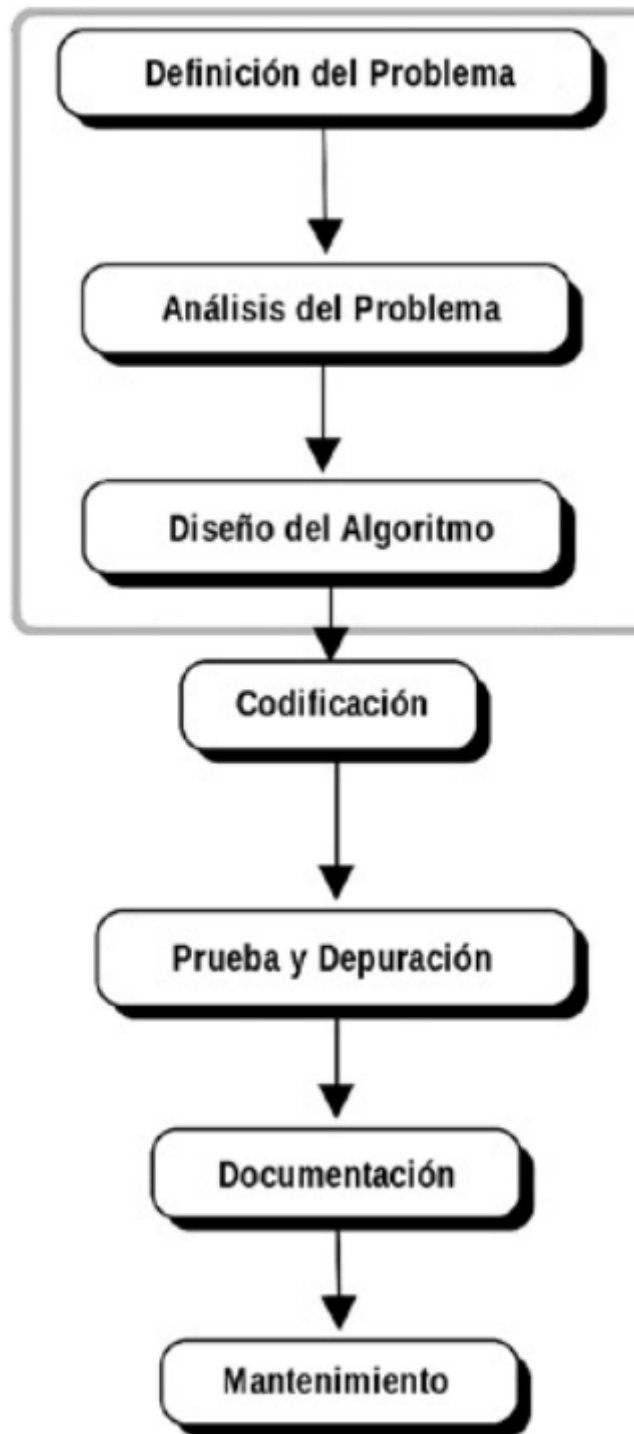


Etapas en la programación



1. Definición del Problema

Primero, necesitamos entender claramente cuál es el problema y qué queremos que haga la computadora. Sin una idea clara, es difícil avanzar, ya que no sabremos bien qué resultados esperamos obtener.

2. Análisis del Problema

Una vez que entendemos el problema, analizamos más a fondo para definir:

- **Datos de entrada:** qué información necesitamos para resolver el problema.
- **Datos de salida:** qué resultado esperamos obtener.
- **Métodos y operaciones:** cómo procesaremos esos datos.

Es útil imaginar que somos la computadora y pensar en qué pasos o instrucciones específicas necesitamos para resolver el problema.

3. Diseño del Algoritmo

En esta fase, creamos un **algoritmo**, que es como una "receta" para resolver el problema. Un algoritmo es:

- **Claro y preciso** (cada paso debe entenderse fácilmente).
- **Finito** (tiene un inicio y un final).
- **Ordenado** (los pasos se siguen en una secuencia lógica).

Entonces, un algoritmo es simplemente un conjunto de pasos específicos que la computadora sigue para resolver un problema. Es como una guía paso a paso para lograr el resultado esperado.

Ejemplos

Problema 1:

Indique la manera de endulzar una taza que contiene café.

- Algoritmo A: Ponerle un poco de azúcar a la taza y revolver
- Algoritmo B: Agregarle una cucharadita de azúcar a la taza, revolver y degustar. Repetir el proceso hasta que quede dulce.

El algoritmo A presenta una solución ambigua al problema planteado. ¿A cuánto equivale un poco? El algoritmo B presenta una solución adecuada al problema.

Problema 2:

Desarrolle un algoritmo que describa la forma de determinar la suma de todos los números naturales.

En este caso, no es posible encontrar un algoritmo que resuelva el problema. Ya que una de las características de un algoritmo es alcanzar la solución en un tiempo finito, situación que no se cumplirá en este caso, ya que los números naturales son infinitos.

Problema 3:

Llenar un pozo con piedras de un bolsón.

Algoritmo: Tomar una pala. Mientras haya piedras en el bolsón cargar la pala con piedras y volcarla en el pozo. Dejar la pala.

Esta solución es un algoritmo, ya que cada paso es no ambiguo. Por otra parte, se puede asegurar que en algún momento finalizará (es finito), aunque no se sabe cuántas paladas serán necesarias.

¿Qué sucedería si no contamos con la pala? ¿Se podría llevar adelante el proceso definido en el algoritmo?

Precondición y postcondiciones

Las **precondiciones** y **poscondiciones** en un algoritmo son condiciones que ayudan a asegurar que funcione correctamente:

- **Precondición:** es lo que sabemos o asumimos como verdadero antes de empezar el algoritmo. Son las condiciones iniciales necesarias para que el algoritmo funcione.
- **Poscondición:** es lo que debe ser verdadero al finalizar el algoritmo, siempre y cuando las precondiciones se hayan cumplido.

En resumen, las precondiciones son las bases para empezar, y las poscondiciones son los resultados esperados al final.

Problema 4:

Determinar el resto de la división entera entre dos números enteros N y M .

Precondición: N y M son números enteros. M debe ser distinto de 0.

Poscondición: el resultado será un valor entero comprendido entre 0 y M-1, que representa el resto de aplicar la división entera entre N y M.

Problema 5:

Indique la manera de endulzar una taza que contiene café.

Precondición: Contar con una cucharita y azúcar suficiente.

Poscondición: El café quedó dulce al finalizar el proceso.

Problema 6:

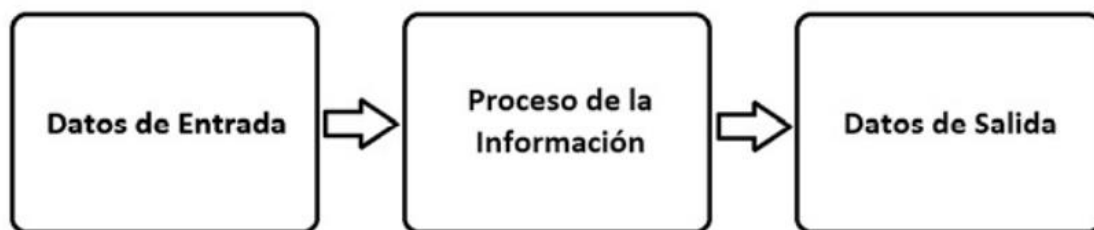
Determinar si el número 437 es primo.

Algoritmo: Dividir el número 437 entre cada uno de los números 1, 2, 3, 4,... 436. Si una de las divisiones es exacta (resto 0), entonces el número 437 no es primo, caso contrario es primo.

Precondición: No hay ningún requerimiento.

Poscondición: Se ha podido determinar si el número 437 es primo o no.

Entrada y salida de datos



Un algoritmo funciona en tres pasos básicos:

1. **Datos de entrada:** es la información que recibe el algoritmo al inicio o mientras se ejecuta.
2. **Procesamiento de datos:** aquí se hacen cálculos o pasos lógicos para resolver el problema.
3. **Salida de resultados:** muestra el resultado final.

En resumen, la computadora no hace nada por sí sola; necesita instrucciones claras (un programa) para resolver un problema. El programador crea el algoritmo como solución al problema, lo traduce a un lenguaje de programación, y luego lo ejecuta en la computadora.