

INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA		CURSO: 10 -	
NOMBRE ESTUDIANTE:		PERIODO: II	

## Guía de Estudio: Introducción a la Programación

### 1. Algoritmo

**Definición:** Un algoritmo es una secuencia finita de instrucciones o pasos que se siguen para resolver un problema o realizar una tarea específica. Es el plan o la estrategia que se utiliza para alcanzar un objetivo determinado.

#### Ejemplo

Algoritmo para preparar un sándwich:

1. Tomar dos rebanadas de pan.
2. Untar mantequilla en una de las rebanadas.
3. Colocar jamón y queso sobre la mantequilla.
4. Colocar la otra rebanada de pan encima.
5. Cortar el sándwich en dos partes.

### 2. Pseudocódigo

**Definición:** El pseudocódigo es una herramienta utilizada para diseñar algoritmos de manera estructurada. Se escribe en un lenguaje natural simplificado, sin preocuparse por la sintaxis de un lenguaje de programación específico. Facilita la comprensión y el diseño de algoritmos antes de codificarlos.

#### Características

- **Simplicidad:** Utiliza un lenguaje sencillo y natural que facilita su comprensión.
- **Claridad:** Es fácil de leer y entender, incluso para quienes no tienen experiencia en programación.
- **Independencia del lenguaje:** No está ligado a ningún lenguaje de programación en particular, lo que lo hace versátil y adaptable.

#### Estructura Básica

El pseudocódigo suele seguir una estructura similar a la de los lenguajes de programación, con elementos como:

INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA		CURSO: 10 -	
NOMBRE ESTUDIANTE:		PERIODO: II	

- **Instrucciones secuenciales:** Se ejecutan en el orden en que aparecen.
- **Condiciones:** Permiten tomar decisiones basadas en ciertas condiciones.
- **Bucles:** Permiten repetir un conjunto de instrucciones varias veces.

### Ejemplo

Pseudocódigo para encontrar el número mayor entre dos números:

**Inicio**

**Leer número1**

**Leer número2**

**Si número1 > número2 entonces**

**Mostrar "Número1 es mayor"**

**Si no**

**Mostrar "Número2 es mayor"**

**Fin**

### Descripción del Ejemplo

1. **Inicio:** Marca el comienzo del algoritmo.
2. **Leer número1 y número2:** Se ingresan dos números.
3. **Condición:** Se compara si número1 es mayor que número2.
  - a) **Si verdadero:** Se muestra que número1 es mayor.
  - b) **Si falso:** Se muestra que número2 es mayor.
4. **Fin:** Marca el final del algoritmo.

### Ventajas del Pseudocódigo

- **Fácil de entender:** No requiere conocimientos avanzados de programación para ser comprendido.
- **Flexibilidad:** Puede ser adaptado fácilmente a diferentes lenguajes de programación.
- **Facilita el diseño:** Permite a los programadores planificar y organizar sus ideas antes de escribir el código.

INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA		CURSO: 10 -	
NOMBRE ESTUDIANTE:		PERIODO: II	

### 3. Diagramas de flujo

**Definición:** Un diagrama de flujo es una representación gráfica de un algoritmo. Utiliza diferentes formas y flechas para mostrar el flujo de las instrucciones y decisiones dentro del proceso. Los diagramas de flujo ayudan a visualizar cómo se debe proceder paso a paso para resolver un problema o ejecutar un proceso específico.

#### Símbolos Comunes

**Óvalo:** Indica el inicio o el fin del proceso.

**Rectángulo:** Representa una instrucción o una acción que se debe realizar.

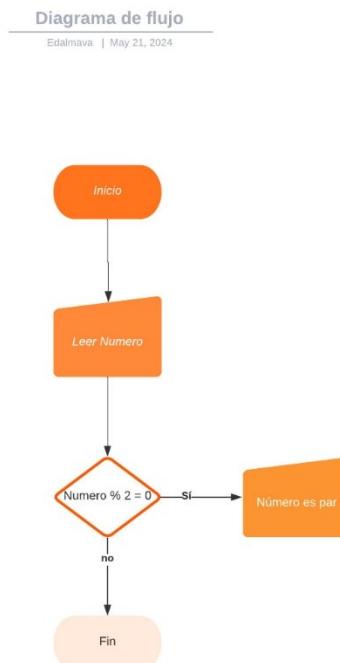
**Diamante:** Indica una decisión que se debe tomar, normalmente una pregunta de sí o no.

**Paralelogramo:** Se utiliza para la entrada o salida de datos (lectura o impresión).

**Flechas:** Muestran la dirección del flujo del proceso.

#### Ejemplo

Diagrama de flujo para decidir si un número es par o impar:



INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA		CURSO: 10 -	
	NOMBRE ESTUDIANTE:		PERIODO: II

## Descripción del Ejemplo

**Inicio:** El proceso comienza.

**Leer número:** Se ingresa el número que se va a evaluar.

**Decisión:** Se verifica si el número es divisible por 2 (Número % 2 == 0).

**Sí:** Si es divisible por 2, el número es par.

**No:** Si no es divisible por 2, el número es impar.

**Mostrar:** Dependiendo del resultado de la decisión, se muestra "Par" o "Impar".

**Fin:** El proceso termina.

## Beneficios

- **Claridad:** Los diagramas de flujo hacen que los procesos complejos sean más fáciles de entender.
- **Comunicación:** Facilitan la comunicación de ideas y procesos entre diferentes personas o equipos.
- **Detección de errores:** Ayudan a identificar posibles errores o ineficiencias en un proceso.

## Uso en Programación

En programación, los diagramas de flujo son especialmente útiles durante la fase de diseño, antes de escribir el código. Ayudan a planificar y visualizar la lógica del programa, asegurando que todas las posibles rutas y decisiones han sido consideradas.

## 4. Variables

**Definición:** Una variable es un espacio en la memoria de un ordenador donde se almacena un valor que puede cambiar durante la ejecución de un programa. Se identifica por un nombre.

### Ejemplo

En un programa que calcula la suma de dos números, `numero1` y `numero2` son variables:

**numero1 = 5**

INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA			CURSO: 10 -
NOMBRE ESTUDIANTE:		PERIODO: II	

**numero2 = 3**

**suma = numero1 + numero2**

**'suma'** es también una variable

## 5. Constantes

**Definición:** Una constante es un valor que no cambia durante la ejecución de un programa. Una vez definida, su valor permanece igual.

### Ejemplo

En un programa que utiliza el valor de  $\pi$  (pi), `PI` es una constante:

**PI = 3.1416**

## 6. Estructuras secuenciales

**Definición:** Las estructuras secuenciales son aquellas en las que las instrucciones se ejecutan una tras otra en el orden en que aparecen.

### Ejemplo

Algoritmo secuencial para sumar dos números:

1. Leer número1
2. Leer número2
3. Sumar número1 y número2
4. Mostrar resultado

## 7. Estructuras condicionales

**Definición:** Las estructuras condicionales permiten tomar decisiones en el flujo de un programa, ejecutando diferentes instrucciones basadas en una condición.

INSTITUCIÓN EDUCATIVA COLEGIO SANTA BÁRBARA			
	GUÍA No 4	FECHA:	
AREA: TECNOLOGIA E INFORMATICA			CURSO: 10 -
NOMBRE ESTUDIANTE:		PERIODO: II	

## Ejemplo

Pseudocódigo para determinar si una persona es mayor de edad:

**Inicio**

**Leer edad**

**Si edad >= 18 entonces**

**Mostrar "Eres mayor de edad"**

**Si no**

**Mostrar "Eres menor de edad"**

**Fin**

## 8. Ejercicios prácticos

### Ejercicio 1: Algoritmo y Pseudocódigo

Escribir el algoritmo y el pseudocódigo para calcular el área de un rectángulo (Área = base \* altura).

### Ejercicio 2: Diagrama de flujo

Crear un diagrama de flujo que represente el proceso de decidir si un número es positivo, negativo o cero.

### Ejercicio 3: Variables y Constantes

Definir un programa en pseudocódigo que calcule la circunferencia de un círculo dado su radio (Circunferencia =  $2 * \pi * \text{radio}$ ). Usa 'PI' como constante.

### Ejercicio 4: Estructuras secuenciales

Desarrollar un algoritmo secuencial en pseudocódigo que convierta una temperatura de grados Celsius a grados Fahrenheit ( $F = C * 9/5 + 32$ ).

### Ejercicio 5: Estructuras condicionales

Escribir un pseudocódigo que determine si un número ingresado por el usuario es primo.