

Introducción a la Programación e Informática

Del pensamiento humano a las máquinas 

Probabilidad y Estadística

Aprendiendo con diversión 

25 de noviembre de 2025

¿Qué vamos a aprender?

💡 Nota importante

¡Bienvenidos al mundo de la programación! 🚀

Objetivo 1: Entender cómo funciona una computadora

Objetivo 2: Conocer la historia de la computación

Objetivo 3: Aprender a pensar como un programador

📝 Ejemplo

Como cuando sigues una receta de cocina paso a paso, programar es dar instrucciones precisas a la computadora

De la Máquina de Turing al Cloud Computing

- 💻 1936: Alan Turing propone la primera computadora teórica
- 💻 1940s: Primeras computadoras electrónicas (ENIAC)
- 💻 1980s: Computadoras personales en los hogares
- 🌐 1990s: Internet y la World Wide Web
- ☁️ 2010s: Cloud Computing y Apps móviles

💡 Nota importante

En menos de 100 años pasamos de máquinas del tamaño de habitaciones a tener supercomputadoras en nuestros bolsillos 📱

Hardware vs. Software

💡 Nota importante

¿Dónde vive y cómo se ejecuta el código?

Hardware 🔧	Software 🖥
Físico y tangible	Digital e intangible
CPU, RAM, disco	Programas y aplicaciones
Se puede tocar	Solo se puede ejecutar
Ej: Teclado, mouse	Ej: Windows, Chrome

📝 Ejemplo

El hardware es como el cuerpo humano, el software es como el cerebro y los pensamientos

El Sistema Binario: El Lenguaje de las Máquinas

Las computadoras solo entienden dos estados: 0 y 1

🔌 0 = Sin electricidad (apagado)

💡 1 = Con electricidad (encendido)

Ejemplo

El número 5 en decimal se representa como 101 en binario

$$101 = 14 + 02 + 11 = 4 + 0 + 1 = 5$$

💡 Nota importante

¡Todo lo que ves en la pantalla son combinaciones de 0s y 1s! ⚡

Representación de Datos en Bits

Bit: La unidad más pequeña (0 o 1)

Byte: 8 bits juntos

Unidad	Equivalencia
1 Byte	8 bits
1 Kilobyte (KB)	1,024 Bytes
1 Megabyte (MB)	1,024 KB
1 Gigabyte (GB)	1,024 MB

Ejemplo

Una letra 'A' ocupa 1 byte, una foto puede ocupar 2 MB

¿Qué es un Lenguaje de Programación?

Es una forma de comunicarnos con la computadora

Los humanos escriben código, la máquina lo ejecuta

💡 Nota importante

Hay cientos de lenguajes: Python **Python**, JavaScript , Java **Java**, C++ 

✍ Ejemplo

`print('Hola Mundo')` → Le dice a la computadora que muestre un mensaje

Lenguajes Compilados vs. Interpretados

Compilados	Interpretados
Se traducen antes ►	Se traducen al momento ►
Más rápidos 🚀	Más flexibles 🔌
C, C++, Java	Python, JavaScript
Ej: .exe en Windows	Ej: Scripts web

Ejemplo

Compilado es como traducir un libro completo antes de leerlo. Interpretado es como tener un traductor en tiempo real

Pensamiento Algorítmico

💡 Nota importante

¡El superpoder de todo programador! 🤖

Algoritmo = Secuencia de pasos para resolver un problema

📝 Ejemplo

Algoritmo para hacer café: 1) Calentar agua, 2) Agregar café, 3) Servir en taza

Los programadores planifican antes de escribir código

Diagramas de Flujo

Son representaciones visuales de algoritmos 

Óvalos: Inicio/Fin

Rombos: Decisiones (Sí/No)

 Rectángulos: Procesos

Flechas: Flujo del programa

Ejemplo

Diagrama para validar login: ¿Usuario existe? → Sí: Permitir acceso / No: Mostrar error

Pseudocódigo: Planificando la Lógica

Es escribir la lógica en lenguaje humano estructurado

Problema

Registro de un nuevo usuario en un sistema

Ejemplo

```
INICIO
    Leer nombre_usuario
    Leer contraseña
    SI contraseña tiene menos de 8 caracteres ENTONCES
        Mostrar "Contraseña muy corta"
    SINO
        Guardar usuario en base de datos
        Mostrar "Usuario registrado exitosamente"
    FIN SI
FIN
```

Ejemplo Práctico: Validación de Edad

① Problema

Crear un algoritmo que valide si una persona puede votar (18+ años)

💡 Ejemplo

```
INICIO
    Leer edad
    SI edad >= 18 ENTONCES
        Mostrar "Puede votar"
    SINO
        Mostrar "No puede votar"
    FIN SI
FIN
```

💡 Nota importante

Este mismo algoritmo se puede escribir en cualquier lenguaje de programación

Aplicación: Sistema de Registro de Usuarios

En un ERP, necesitamos registrar empresas y empleados

Ejemplo

Pseudocódigo para validar RUT chileno:

1. Leer RUT ingresado
2. Verificar formato (XX.XXX.XXX-X)
3. Calcular dígito verificador
4. Comparar con el dígito ingresado
5. Si coincide, RUT válido
6. Si no coincide, RUT inválido

¡Tu Turno de Practicar!

➊ Problema

Crea un algoritmo para aprobar o rechazar una transacción bancaria

Paso 1: Leer el saldo actual de la cuenta

Paso 2: Leer el monto de la transacción

Paso 3: Comparar si el saldo es mayor o igual al monto

Paso 4: Decidir aprobar o rechazar

💡 Nota importante

Piensa en la lógica antes de escribir código. ¡Ese es el pensamiento algorítmico!

¡Resumen de lo Aprendido!

- ✓ La historia de la computación nos muestra cómo llegamos hasta aquí
- ✓ Hardware es físico, Software es lógico
- ✓ Las máquinas hablan en binario (0 y 1)
- ✓ Los lenguajes de programación traducen nuestras ideas a código
- ✓ Los algoritmos son planes paso a paso para resolver problemas
- ✓ Diagramas de flujo y pseudocódigo nos ayudan a planificar

Nota importante

¡Ya tienes las bases para empezar a programar! 

¡Sigue aprendiendo! 