

# Introducción a los Lenguajes de programación

Profesor: Matías Greco y  
Nicolás Hormazábal

# Código de honor

---

**Toda persona inscrita en este curso se compromete a:**

- **Actuar con honestidad, rectitud y buena fe frente a sus profesores y compañeros.**
- **No presentar trabajos o citas de otras personas como propias ni sin su correspondiente citación, ya sea de algún compañero, libro o extraídos de internet; como también a no reciclar trabajos presentados en semestres anteriores como trabajos originales.**
- **No copiar a compañeros ni hacer uso de ayudas o comunicaciones fuera de lo permitido durante las evaluaciones.**



# Lenguaje

---

## Que significa lenguaje?

- Facultad del ser humano de expresarse y comunicarse con los demás a través del sonido articulado o de otros sistemas de signos (RAE).

# Lenguaje de programación

---

## Que es un lenguaje de programación?

- Un conjunto de caracteres, reglas para combinarlos y reglas que especifiquen sus efectos cuando son ejecutados por un computador.

# Lenguaje de máquina

---

## Que es el lenguaje de máquina?

- Es el lenguaje nativo de una computadora.
- Sus instrucciones son secuencias de 0 y 1 (*patrones de bit*, tales como 11110000, 01110011...)
- Sin embargo, las computadoras sólo entienden las instrucciones en lenguaje máquina, por lo que será preciso traducir los programas resultantes a lenguajes de máquina antes de que puedan ser ejecutadas por ellas.



# Lenguaje de programación

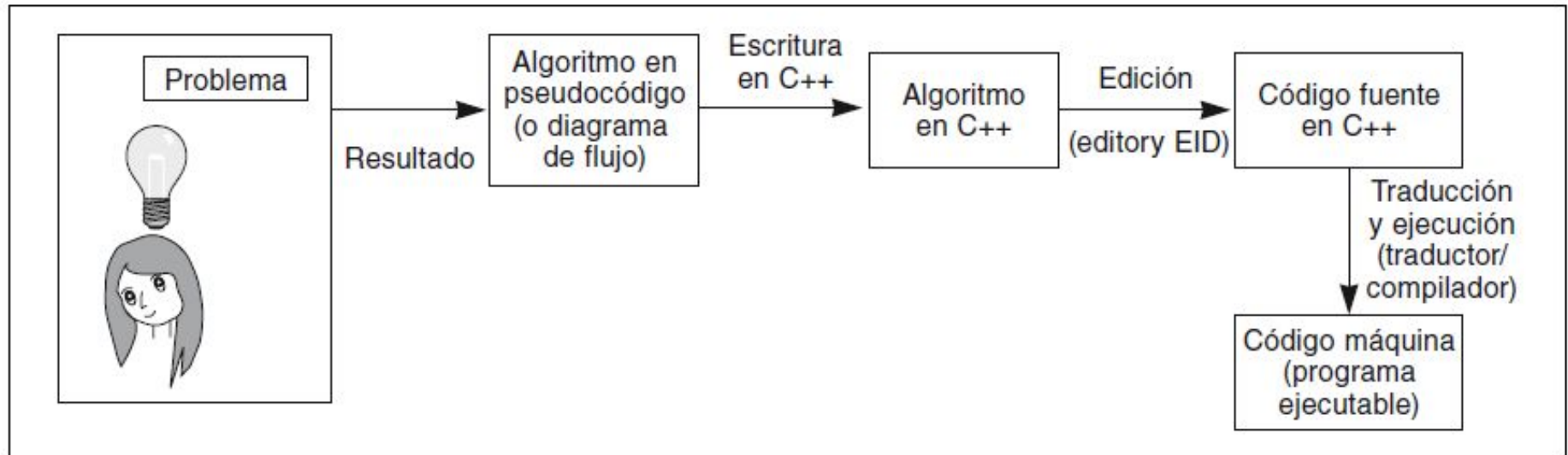
---

## **Características de un lenguaje de programación?**

- No requiere conocimiento del código de máquina.
- Se traduce a lenguaje de máquina
- Emplea una notación que está más cerca de aquella usada para el problema específico que se resuelve que del código de máquina



# Lenguaje de programación



**Figura 1.14.** Proceso de transformación de un algoritmo en pseudocódigo en un programa ejecutable.

Fuente: (Joyanes, 1988)

# Traductores de lenguaje

---

- Se le llama así al proceso de conversión de un lenguaje de alto nivel a código de máquina.
- Se dividen en dos tipos: compiladores e intérpretes.



# Intérpretes

---

- Un *intérprete* es un traductor que toma un programa fuente, lo traduce y, a continuación, lo ejecuta.
- Ejemplos



# Compiladores

---

- Un *compilador* es un programa que traduce los programas fuente escritos en lenguaje de alto nivel a lenguaje máquina.
- La traducción del programa completo se realiza en una sola operación denominada **compilación** del programa; es decir, se traducen todas las instrucciones del programa en un solo bloque, generando un “programa ejecutable”.

# Compiladores

---

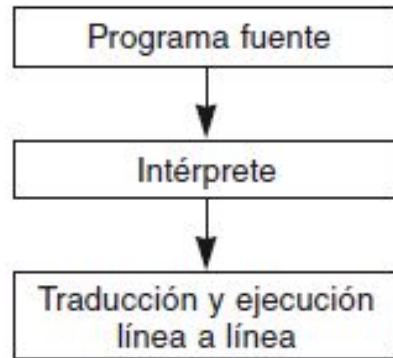


Figura 1.16. Intérprete.

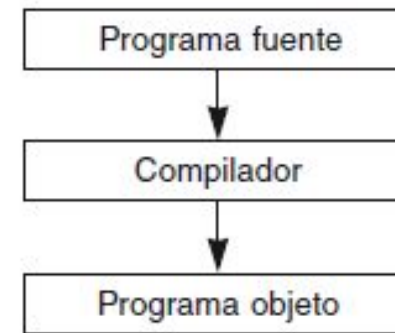
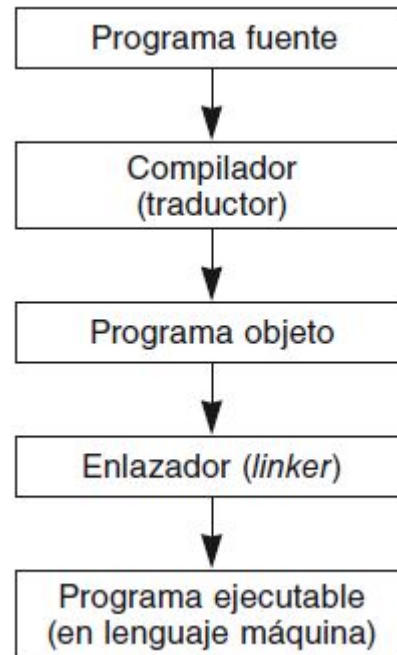


Figura 1.17. La compilación de programas.

# Compiladores

---



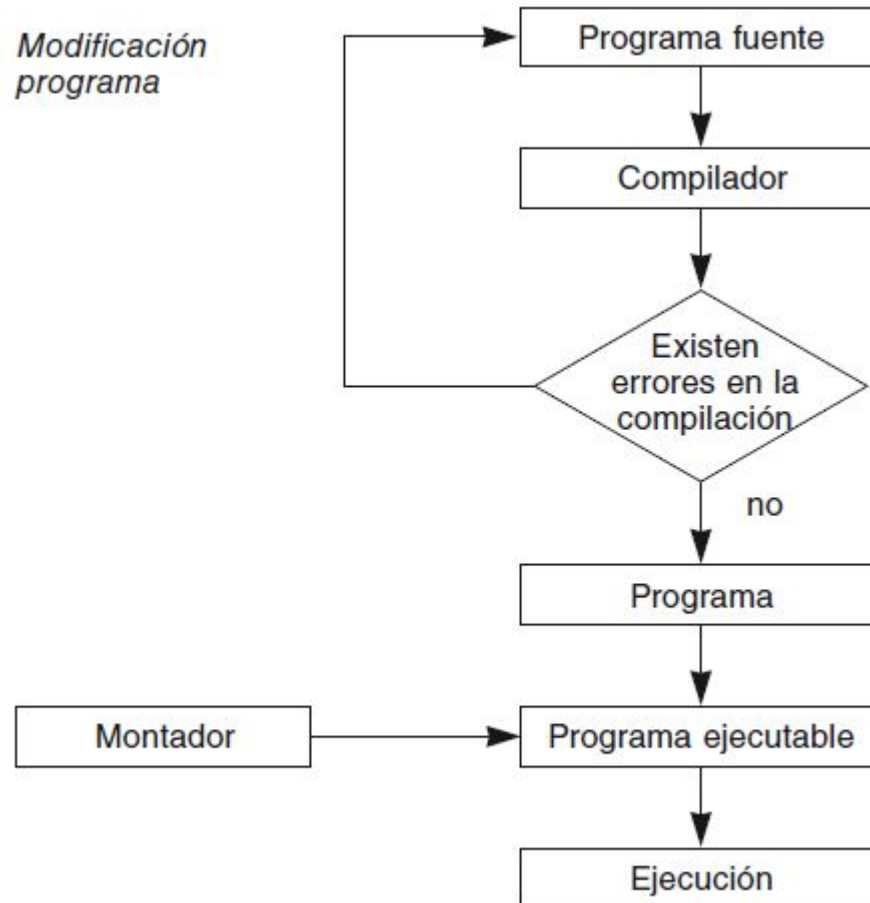
Fases de compilación.  
Fuente: (Joyanes, 1988)

# Investigue

---

- Averiguar qué son los lenguajes híbridos y cómo funciona el Lenguaje Java.
- Involucre el concepto de “lenguaje intermedio”

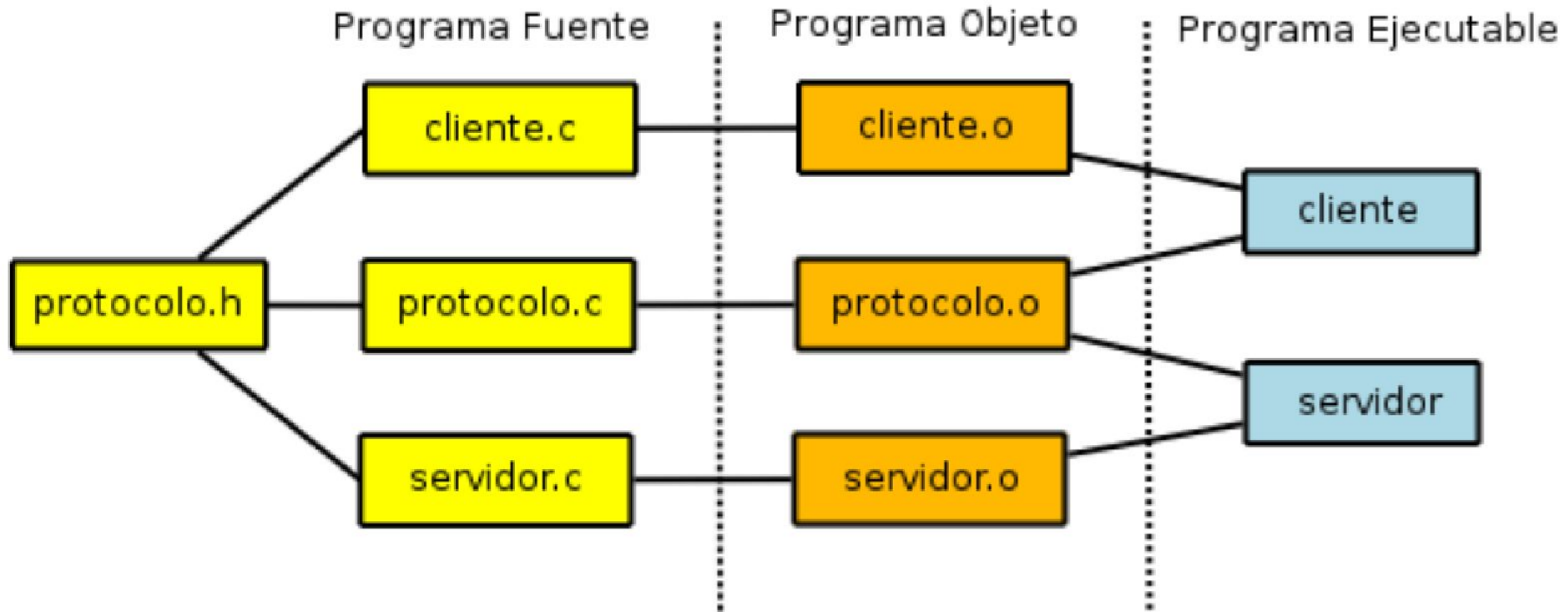
# Compiladores



Fases de ejecución de un programa.  
Fuente: (Joyanes, 1988)

# Compiladores

Ejemplo en C



# Lenguajes de programación

---

- Lenguajes de primera generación (lenguaje de máquina)
- Lenguajes de segunda generación (lenguajes de bajo nivel)
- Lenguajes de tercera generación
- Lenguajes de cuarta generación



# Lenguajes de segunda generación

---

- Ensambladores (assemblers)
  - Desde 1950.
  - Simplifica el proceso de escribir código de máquina.
  - Utiliza un formato nemotécnico en vez de formatos numéricos.

# Lenguajes de segunda generación

---

- Ensambladores (assemblers)

Mover el contenido del registro 4 al registro 28

4 0 4 8                      0 0 1 0   0 0 0 0   0 0 1 0   1 0 0 0

Lenguaje de máquina

MOV R5, R6

Ensamblador



# Lenguajes de tercera generación

---

- Sus instrucciones son de alto nivel (comprensibles por el programador)
- Independientes de la máquina.
- Ejemplos: C, C++, Java, Fortran.

# Lenguajes de cuarta generación

---

- Herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computador)
- MDD (Model driven development)



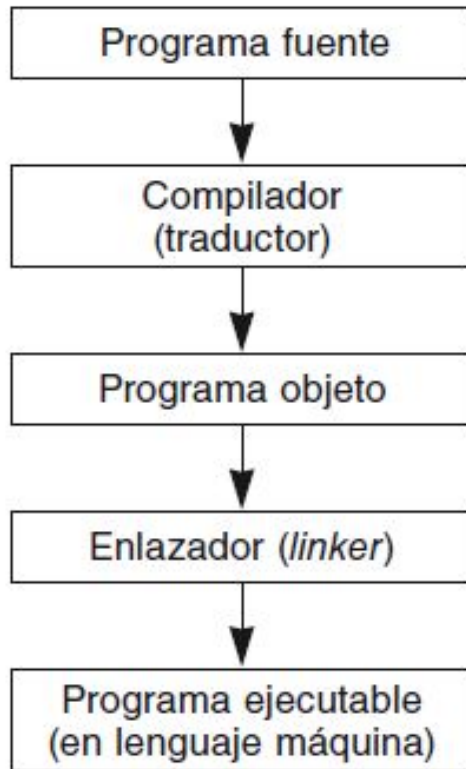
# Biblioteca

---

- Recopilación de cabeceras y funciones.
- Agrupan funciones por el tipo de trabajo que hacen (Matemáticas, entrada y salida, base de datos, etc)

# Compilación y enlazado

---



Para más detalles de este proceso en lenguaje C utilizando el compilador GCC, visitar la siguiente URL:

<https://iie.fing.edu.uy/~vagonbar/gcc-make/gcc.htm#2.Compilacion>



# Recapitulación

---

- Clasificación de los lenguajes de programación por:
  - Compilado o interpretado
  - Generación (evolución)
  - Paradigma

# Paradigmas

---

- Conjunto de prácticas que definen una disciplina científica durante un periodo específico de tiempo. (Thomas Khun)
- Ejemplo o ejemplar. Patrón.



# Paradigmas

---

“En realidad un **paradigma de programación** representa fundamentalmente enfoques diferentes para la construcción de soluciones a problemas y por consiguiente afectan al proceso completo de desarrollo de software”  
(Joyanes, 1988).



# Paradigmas

---

- Imperativo
- Declarativo
- Orientado a objeto
- Concurrente

# Paradigmas

---

- **Imperativo**

- Engloba paradigma estructurado, procedimental y modular.
- Es un conjunto de instrucciones que se ejecutan una por una, de principio a fin, de modo secuencial.
- Define el proceso de programación como el desarrollo de una secuencia de órdenes (*comandos*) que manipulan los datos para producir los resultados deseados.
- En un lenguaje procedimental cada instrucción es una orden u órdenes para que la computadora realice alguna tarea específica

# Paradigmas

---

- **Declarativo**

- Solicita al programador que describa el problema en lugar de encontrar una solución algorítmica al problema.
- Un lenguaje declarativo utiliza el principio del razonamiento lógico para responder a las preguntas o cuestiones consultadas.
- Se basa en la *lógica formal* y en el *cálculo de predicados de primer orden*.



# Paradigmas

---

- **Orientado a objetos**

- Es análogo a la vida real.
- Se basa en el diseño y construcción de objetos, que se componen a su vez de datos y operaciones que manipulan esos datos.
- El programador define en primer lugar los objetos del problema y a continuación los datos y operaciones que actuarán sobre esos datos.
- Derivan esencialmente de la estructura modular existente en la vida real y el modo de respuesta de estos módulos u objetos a mensajes o eventos que se producen en cualquier instante.
- Claves: Encapsulamiento, herencia y polimorfismo.

# Paradigmas

---

- **Concurrentes**

- Permite la utilización de más de un proceso o hilo a la vez, de manera que se puedan comunicar para obtener una solución. Estos procesos o hilos se encuentran compitiendo por los recursos del computador.

# Paradigmas

---

- **Clasifique los siguientes lenguajes de programación según los paradigmas vistos:**
  - C
  - C++
  - Java
  - PHP
  - Matlab

# Introducción a los Lenguajes de programación

Profesor: Nicolás Hormazábal y  
Matías Greco