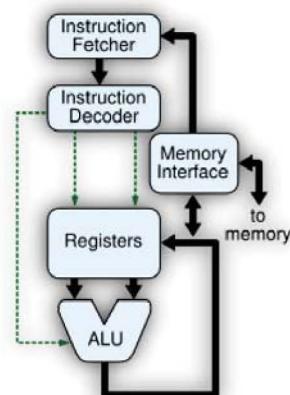


Tema 1: El Programa Informático

Tema 1: El Programa Informático
Entornos de Desarrollo

Alejandro Cardo Grau



Entornos de Desarrollo

T1: El Programa Informático
ÍNDICE

CRONOLOGÍA 1º/2º TRIMESTRE

		Capítulos del Libro	
BLOQUE I: INTRODUCCIÓN A LOS ENTORNOS DE DESARROLLO	T1: El Programa Informático	CAPÍTULO 1. DESARROLLO DE SOFTWARE	Pendiente para el 2º trimestre: 1.6 ARQUITECTURA DE SOFTWARE 1.6.1 Patrones de desarrollo 1.6.2 Desarrollo en tres capas
	T2: Entornos de Desarrollo	CAPÍTULO 2. INSTALACIÓN Y USO DE ENTORNOS DE DESARROLLO	
BLOQUE II: ANÁLISIS Y DISEÑO DE PROYECTOS	T3: Ingeniería del Software	CAPÍTULO 1. DESARROLLO DE SOFTWARE	
	T4: Modelado del Proceso	CAPÍTULOS 5 y 6. DISEÑO ORIENTADO A OBJETOS. DIAGRAMAS DE COMPORTAMIENTO	5.1 INTRODUCCIÓN A UML 6.1 Tipos y campo de aplicación 6.2 Diagramas de actividad
	T5: Modelado de Análisis de Requisitos	CAPÍTULO 6. DISEÑO ORIENTADO A OBJETOS. DIAGRAMAS DE COMPORTAMIENTO	6.3 Diagramas de casos de uso

Alejandro Cardo Grau

Entornos de Desarrollo

- 1. El Programa Informático**
- 2. Lenguajes de Programación**
 - 1. Características de los lenguajes de programación**
 - 2. Clasificación de los lenguajes de programación**
- 3. Obtención del Código Ejecutable**
- 4. Referencias**

1. EL PROGRAMA INFORMÁTICO

Definición de un programa informático:

- Es un conjunto de instrucciones que se ejecutan de manera secuencial con el objetivo de realizar **una o varias tareas** en un sistema informático.

Características de un programa informático:

- **Un programa interactúa con el sistema informático dividiendo estas tareas y ejecutándolas como diferentes instrucciones del programa dentro de la Unidad Aritmética Lógica (ALU):**

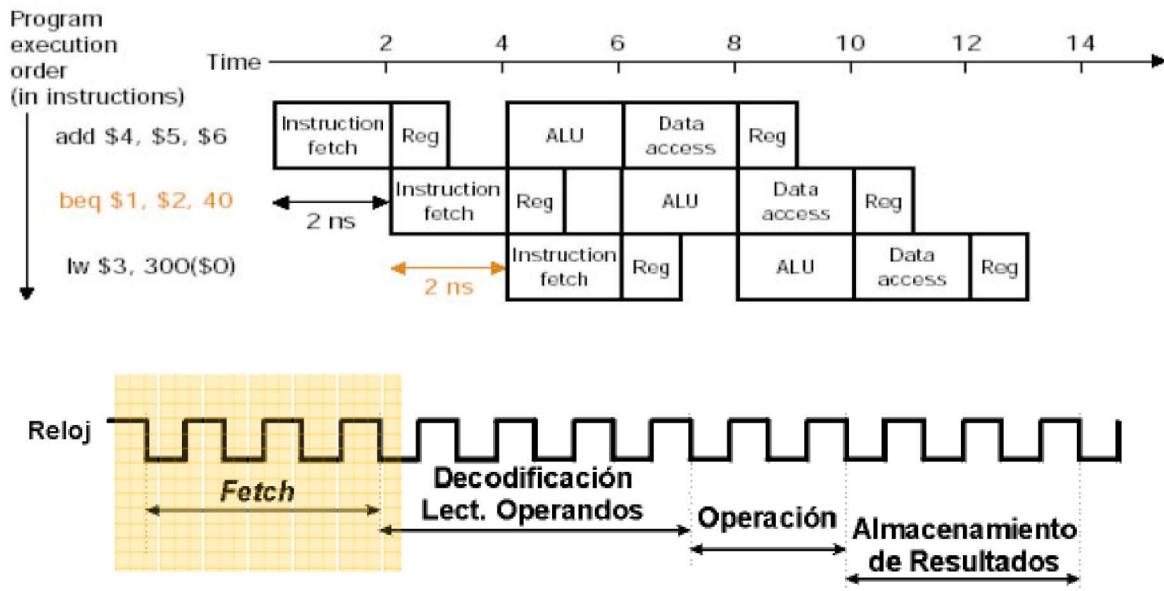
- La ALU, junto a la Unidad de Control (UC) encargado de recibir las instrucciones, es una parte del microprocesador o **Unidad Central de Proceso (CPU)**.
- Actualmente los microprocesadores multinúcleo pueden tener múltiples ALU para la resolución de potentes cálculos complejos.

T1: El Programa Informático

1. EL PROGRAMA INFORMÁTICO

- **Características de la ALU:**

- Cada instrucción se divide en instrucciones más pequeñas (**microinstrucciones**) que se ejecutarán individual y secuencialmente en cada ciclo del microprocesador.



Alejandro Cardo Grau

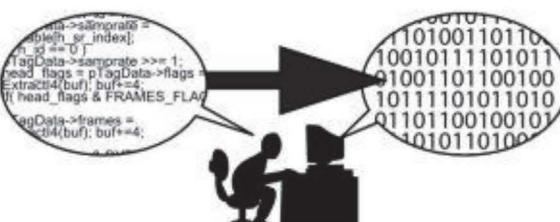
Entornos de Desarrollo

T1: El Programa Informático

2. LENGUAJES DE PROGRAMACIÓN

Definición de un lenguaje de programación:

- Es un **conjunto de instrucciones, operadores y reglas de sintaxis y semánticas**, que se ponen a disposición del programador para que éste pueda comunicarse con los dispositivos de hardware y software existentes.
- El sistema sólo es capaz de entender código escrito en código o lenguaje máquina (1s y 0s).
- Programar directamente en código máquina es una tarea ardua y tediosa y altamente dependientes del tipo de máquina.



Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

Clasificación de los lenguajes de programación:

- **Nivel de abstracción:**

- Bajo nivel / lenguaje máquina
- Medio nivel
- Alto nivel y muy alto nivel

- **Forma de ejecución:**

- Lenguajes compilados
- Lenguajes interpretados
- Lenguajes virtuales

- **Paradigma de programación:**

- Imperativo
- Declarativo

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

Nivel de abstracción:

- **El uso de un lenguaje de programación tiene el objetivo de facilitar la escritura de programas utilizando un mayor nivel de abstracción en el código:**

- El nivel de abstracción de un lenguaje implica cuanto está alejado del código o lenguaje máquina.
- Cuanto más parecido sea a nuestro lenguaje natural (sea más entendible), el lenguaje se considerará de **lenguaje de programación de alto nivel** (Ejemplos: Java, C#).

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

- Bajo nivel:** Sólo hay un lenguaje de primera generación, el código o lenguaje máquina.

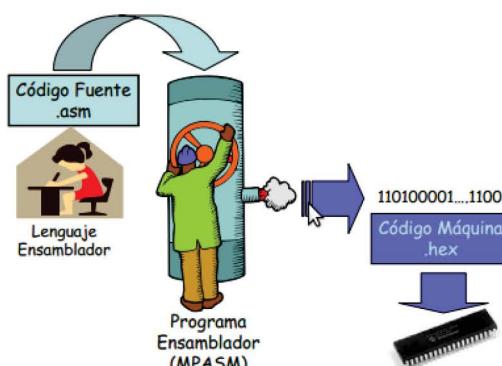
```
-u 100 1a
OCFD:0100 BAQB01
OCFD:0103 B409
OCFD:0105 CD21
OCFD:0107 B400
OCFD:0109 CD21
-d 10b 13f
OCFD:0100 20 65 73 74 65 20 65 73-20 75 48 6F 6C 61 2C
OCFD:0110 72 61 6D 61 20 68 65 63-68 6F 20 65 6E 20 61 73
OCFD:0120 73 65 6D 62 6C 65 72 20-70 61 72 61 20 6C 61 20
OCFD:0130 57 69 6B 69 70 65 64 69-61 24
OCFD:0140
```

Hola,
este es un progra
rama hecho en as
sembler para la
Wikipedia\$

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

- Medio nivel:** Tienen definidas una serie de instrucciones sencillas para trabajar con datos simples y posiciones de memoria.
 - Requieren conocimiento de la arquitectura del microprocesador.
 - Repertorio limitado exclusivo de instrucciones dependientes del tipo de arquitectura (CISC, RISC).
 - El lenguaje de programación es el **lenguaje ensamblador**.



- Existen lenguajes a medio-alto nivel (intermedios) que permiten incorporar rutinas en ensamblador. Ejemplo: Lenguaje C. → **CÓDIGO EMBEBIDO**

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

```

;*****+
;Programa E001.asm           Fecha: 3 Diciembre 2004
;Este programa suma dos valores inmediatos (7+8) y el resultado
;lo deposita en la posición 0x10
;Revisión: 0.0                 Programa para PIC16F84
;Velocidad de reloj: 4 MHz      Instrucción: 1Mz=1 us
;Perro Guardián: deshabilitado Tipo de Reloj: XT
;Protección de código: OFF
;*****+
LIST    p=16F84 ;Tipo de PIC
;*****+
RESULTADO EQU 0x10 ;Define la posición del resultado
;*****+
ORG 0 ;Comando que indica al Ensamblador
;la dirección de la memoria de programa
;donde situar la siguiente instrucción
;*****+
INICIO    movlw   0x07 ;Carga primer sumando en W
          addlw   0x08 ;Suma W con segundo sumando
          movwf   RESULTADO ;Almacena el resultado
          END      ;Fin del programa fuente
;*****+

```

Programa realizado en lenguaje ensamblador de suma de dos números para un microcontrolador RISC

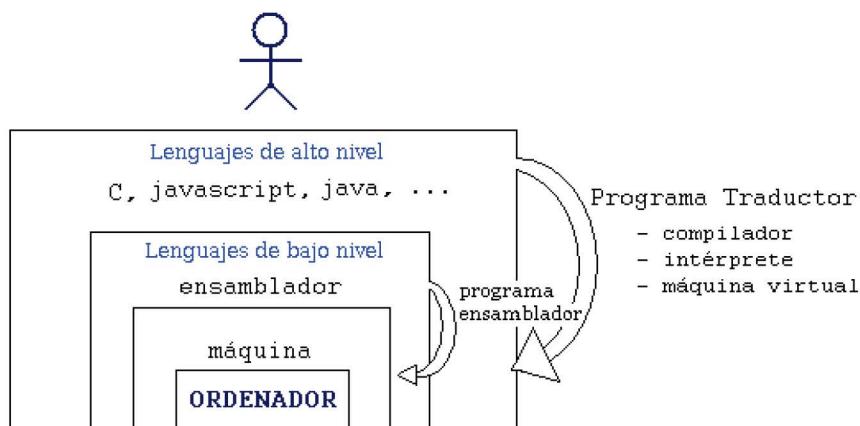
Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.1 Características de los lenguajes de programación

- Alto nivel:** La mayoría de los lenguajes que se utilizan hoy en día para programar aplicaciones son de alto nivel.
- Son muy cercanos a nuestro propio lenguaje y permiten la realización de patrones de diseño complejos como **Java** o **C#**.
- También nos encontramos en éste nivel de abstracción a los lenguajes destinados a un propósito específico, como podría ser el lenguaje **SQL** para la gestión/administración de bases de datos.
- Dos grandes estilos o **paradigmas:** **imperativo** y **declarativo**.



Alejandro Cardo Grau

Entornos de Desarrollo

- **PREGUNTA 1:**

- ¿A qué nivel de los lenguajes de programación asociarías un lenguaje donde cada instrucción corresponde a una acción determinada a realizar por la CPU?

- **PREGUNTA 2 (TEST):**

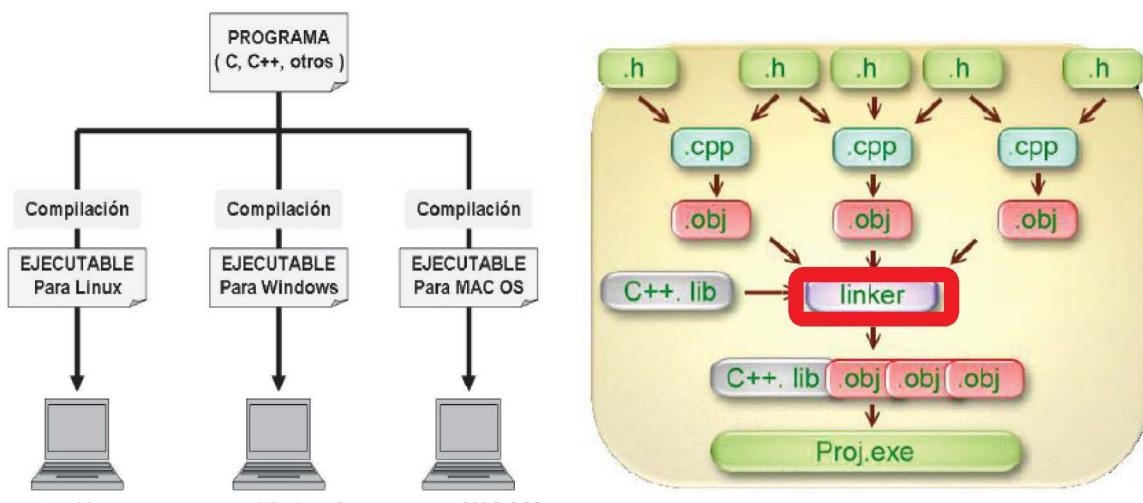
- Indicar cuál de las siguientes afirmaciones es cierta:

- 1) Programar directamente en código máquina es una tarea compleja.
- 2) Cuanto más parecido sea el lenguaje de programación a nuestro lenguaje natural será mas abstracto y de mayor nivel.
- 3) Cada microinstrucción de un programa informático se ejecuta individualmente y secuencialmente en cada ciclo del microprocesador
- 4) Todas las anteriores son correctas.

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- **Compilados:** Un programa traductor (compilador) convierte el código fuente en código objeto para crear un programa ejecutable (.exe).
- Ejemplos: C, C++, Objective-C.



Proceso de Compilación y Ficheros Relacionados en el Lenguaje C/C++

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- **Interpretados:** Ejecutan las instrucciones directamente traduciéndolas, sin generar código objeto. La propia máquina se encargará de interpretarlo instrucción a instrucción usando un tipo de traductor: intérprete.
 - Ejemplos: PHP, Javascript ó Phyton.



Interpretación de un PHP script en un servidor web de Internet

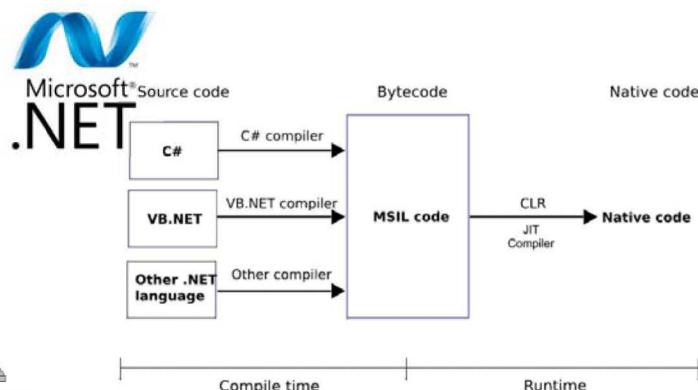
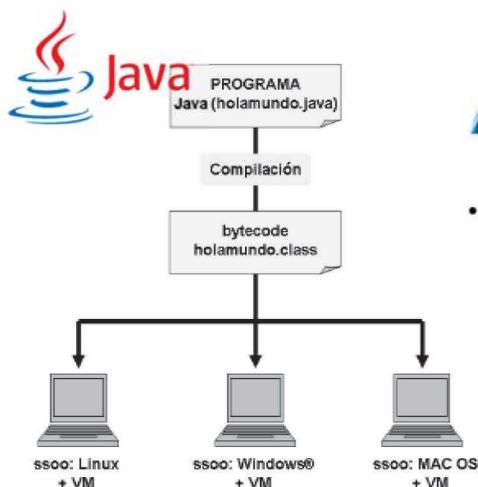
Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- **Virtuales (Mixtos):** Con un comportamiento similar al de los lenguajes compilados con una peculiaridad, el compilador no genera código objeto sino código *bytecode*.
 - Una máquina virtual (intérprete), instalada en el S.O., se encargará de interpretar el *bytecode* para ejecutar el programa.
 - Suelen ser ejecutados en entornos multiplataforma.
 - Ejemplos: Java, Visual Basic .NET, C# (framework .NET)



Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- **Imperativo:** Describe la programación como una secuencia de instrucciones que cambian el estado del programa, indicando cómo realizar una tarea.
 - Ejemplos: Java, Javascript, C, C++, C#, Phyton, PHP
- **El paradigma imperativo se divide en dos tipos:**
 - **Procedimental:** El programa se divide en partes más pequeñas, llamadas **funciones y procedimientos**, que pueden comunicarse entre sí.
 - Permite reutilizar código ya programado y solventa el problema de la **programación spaghetti**
 - Flujo del código sin estructuración y uso de saltos incondicionales: GOTO's.
 - **Orientado a objetos:** Encapsula el estado y las operaciones en objetos, creando una estructura de clases y objetos que emula un modelo del mundo real.

Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- **Declarativo:** Especifica o declara un conjunto de premisas y condiciones para indicar QUÉ es lo que hay que hacer y no necesariamente CÓMO hay que hacerlo.
 - Ejemplos: SQL, Haskell, Prolog
- **El paradigma declarativo se divide en dos tipos:**
 - **Funcional:** Evalúa el problema realizando funciones de manera recursiva, evita declarar datos y haciendo hincapié en la composición de las funciones y en las interacciones entre ellas
 - **Lógico:** Define un conjunto de reglas lógicas para ser interpretadas mediante inferencias lógicas. Permite responder preguntas planteadas al sistema para resolver problemas.

Principales lenguajes de programación:

<http://www.larevistainformatica.com/LENGUAJES-DE-PROGRAMACION-listado.html>

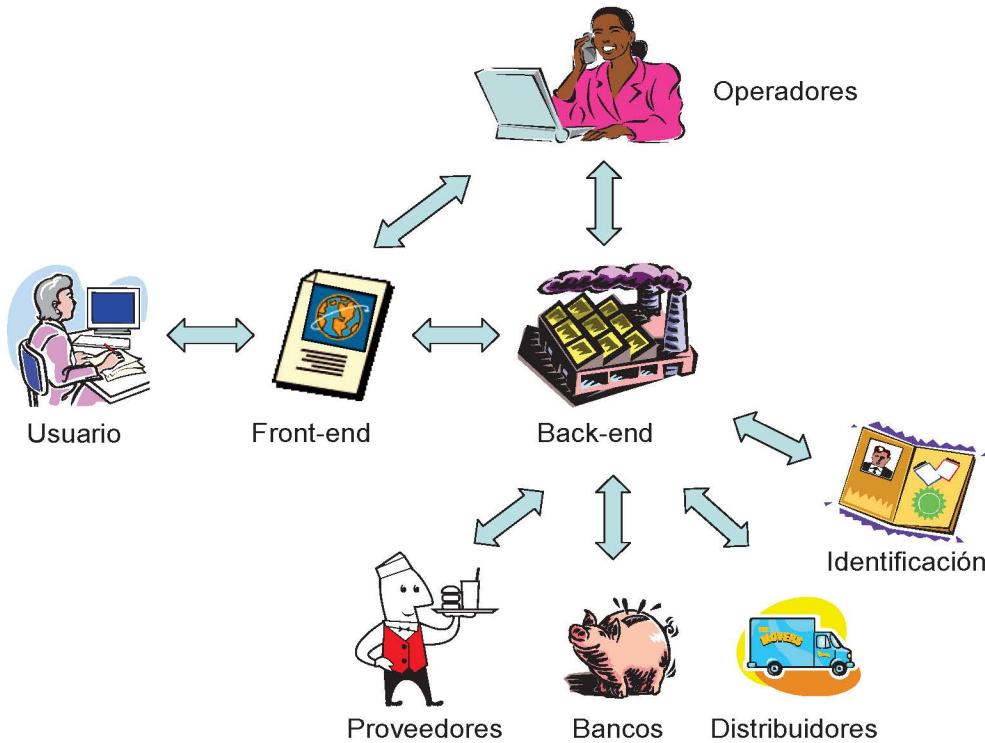
Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

- ¿Qué tecnologías y lenguajes de programación debe conocer un programador web actual?



Alejandro Cardo Grau

Entornos de Desarrollo

2. Lenguajes de Programación

2.2 Clasificación de los lenguajes de programación

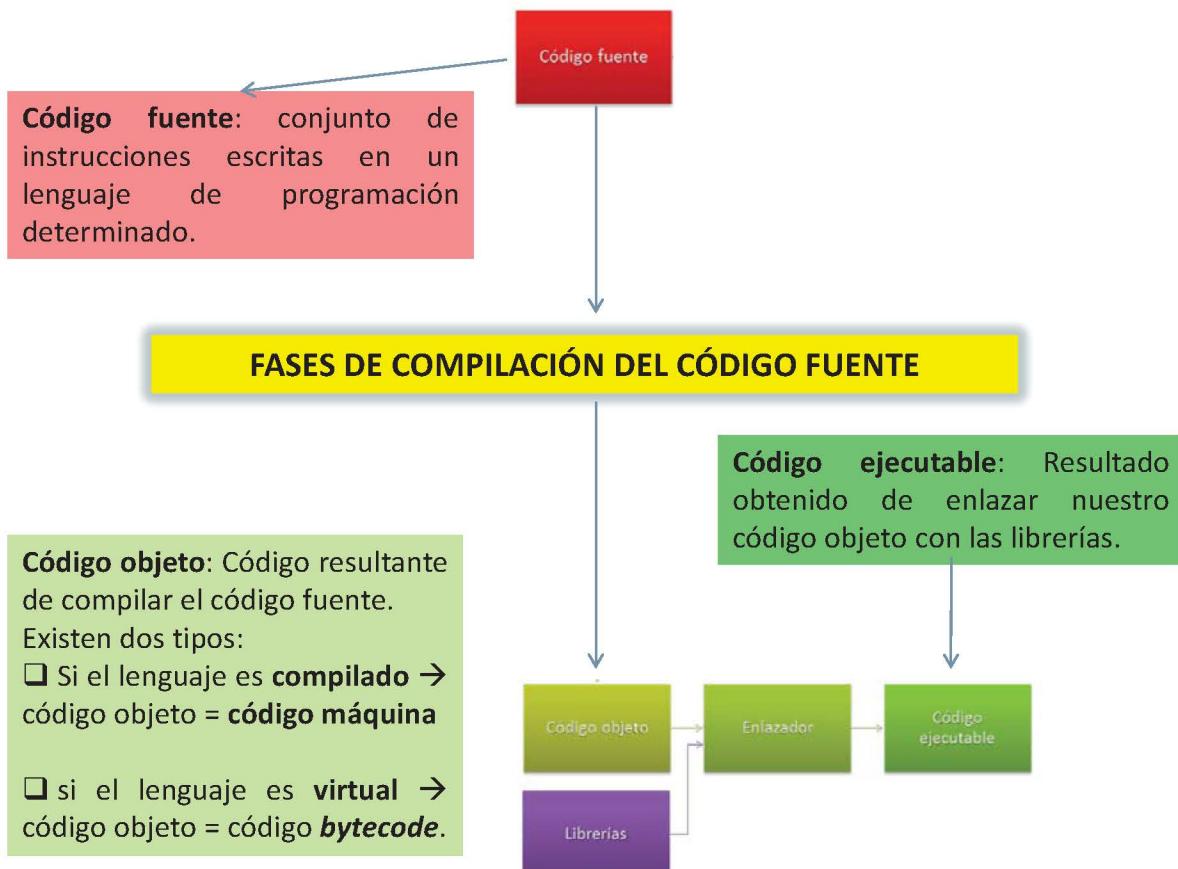
- ¿Qué tecnologías y lenguajes de programación debe conocer un programador web actual?
 - **FRONTEND** (lado del cliente):
 - El lenguaje HTML (*Hypertext Markup Language*)
 - El uso de hojas de estilo CSS (*Cascading Style Sheets*)
 - Lenguajes orientados al lado del cliente (Javascript/JQuery Mobile)
 - Manejo de la Jerarquía de objetos DOM (Document Object Model)
 - **BACKEND** (lado del servidor):
 - Un lenguaje orientado al lado del servidor (PHP, Django, Node.js, Phyton)
 - Administración y programación de gestores de contenido (Drupal,Django)
 - El manejo de operaciones en base de datos de servidor (SQL / Oracle)
 - Características de configuración de servidores web (Apache / Tomcat)
 - Procesamiento de la información con XML (Java / C# / XPath / XQuery)
 - **ENTORNOS DE DESARROLLO:**
 - Netbeans
 - Eclipse
 - Otros IDE especializados

Alejandro Cardo Grau

Entornos de Desarrollo

T1: El Programa Informático

3. OBTENCIÓN DEL CÓDIGO EJECUTABLE

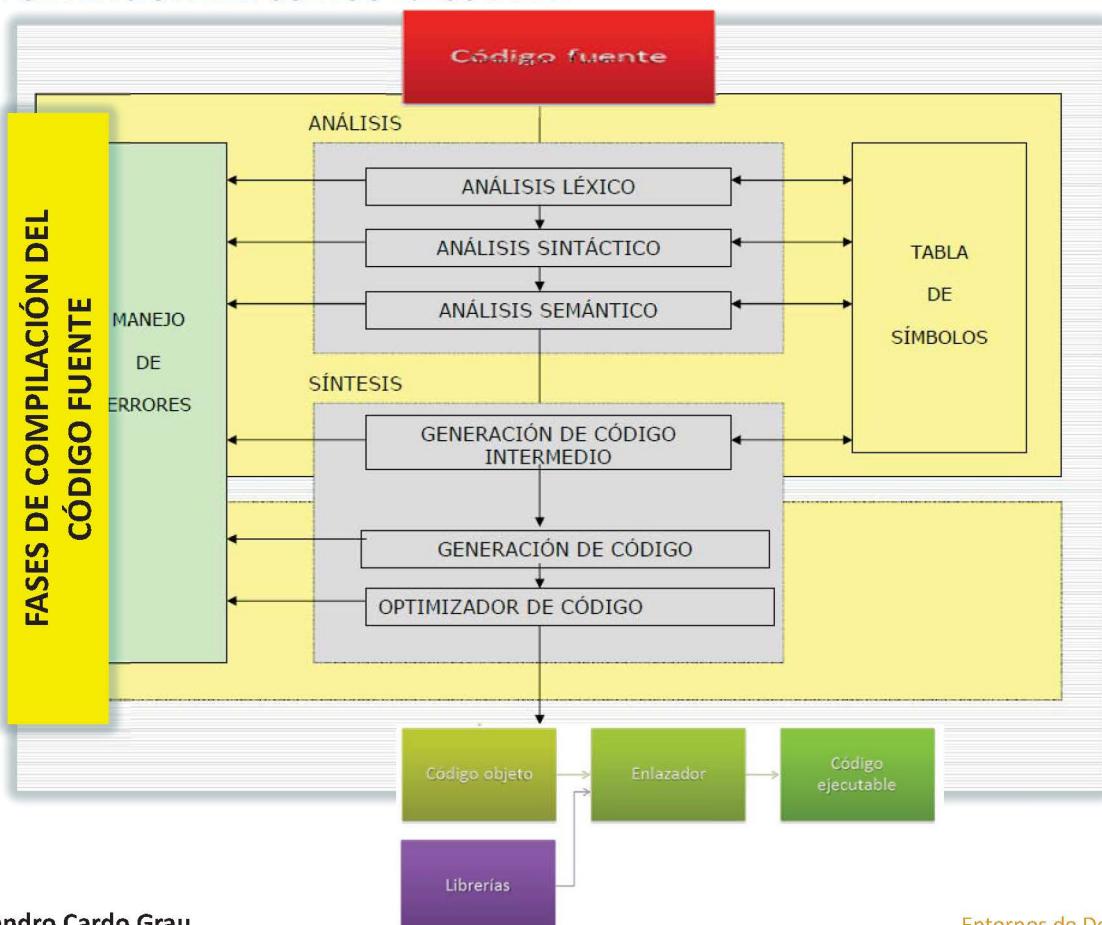


Alejandro Cardo Grau

Entornos de Desarrollo

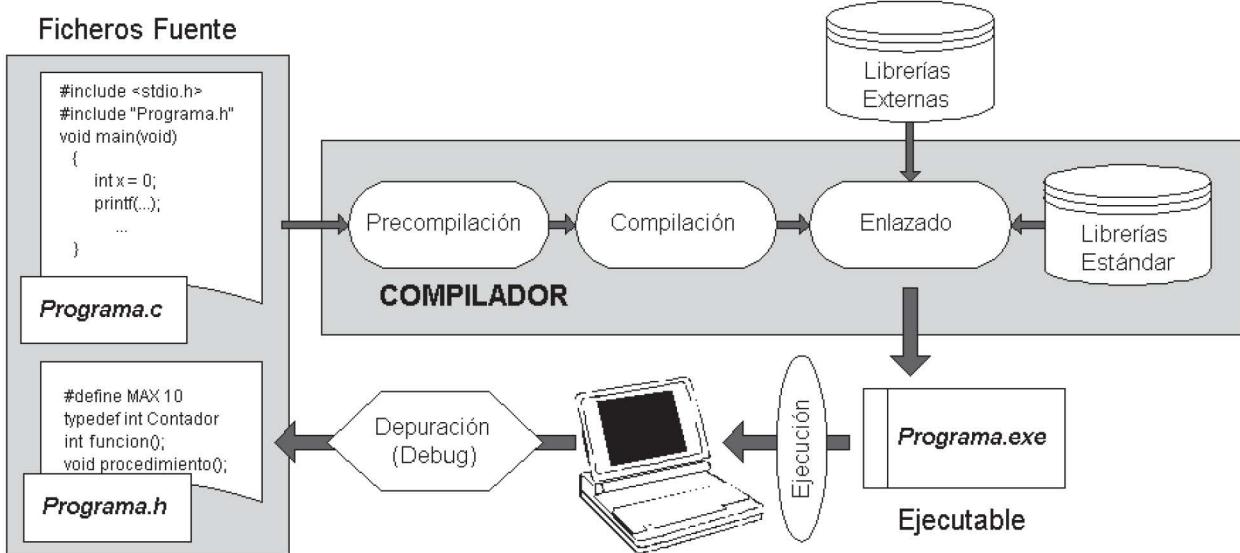
T1: El Programa Informático

3. OBTENCIÓN DEL CÓDIGO EJECUTABLE



Alejandro Cardo Grau

Entornos de Desarrollo

3. OBTENCIÓN DEL CÓDIGO EJECUTABLE**Obtención del código ejecutable del lenguaje C**

Alejandro Cardo Grau

Entornos de Desarrollo

REPASO

- **PREGUNTA 1 (TEST):**
 - PHP es un lenguaje de programación *back-end* que procesa y traduce las instrucciones en tiempo de ejecución y, por tanto, es:
 - 1) Compilado
 - 2) Interpretado
 - 3) Virtual
 - 4) Ninguna de las otras respuestas son correctas.
- **PREGUNTA 2 (TEST):**
 - ¿Qué elemento del proceso de compilación permite unir archivos en la generación del ejecutable?:
 - 1) Código fuente
 - 2) Intérprete
 - 3) Ensamblador
 - 4) Ninguna de las otras respuestas son correctas.

T1: El Programa Informático

4. REFERENCIAS



Alejandro Cardo Grau

Entornos de Desarrollo

T1: El Programa Informático

4. REFERENCIAS

- Casado Iglesias, C. (2012). *Entornos de Desarrollo*. CFGS. Editorial Ra-Ma.
- *Entornos de Desarrollo (IDE)*: <http://programacion-laura.blogspot.com.es/2011/08/entorno-de-desarrollo-integrado-ide.html>
- *Nuevas profesiones, aquí si hay empleo*: <http://www.abc.es/20120229/economia/abci-nuevas-profesiones-futuro-empleo-201202281538.html>

Alejandro Cardo Grau

Entornos de Desarrollo