

Universidad de Los Andes
Facultad de Ingeniería
Escuela de Ingeniería de Sistemas
Departamento de Computación

PROGRAMACIÓN 1

Sección 02
A-2012

Ícaro Alzuru Cortez
icaro@ula.ve

HORARIO

Teoría (y práctica):

Lunes: 4-6 pm (**LCNW** o 4O-13)

Miércoles: 4-6 pm (**LCNW** o 4O-13)

Laboratorio:

Jueves: 2-4 pm (LCNW)

<http://webdelprofesor.ula.ve/ingenieria/icaro>

EVALUACIÓN

1er Parcial:	15%	(02/05/2012)
2do Parcial:	20%	(30/05/2012)
3er Parcial:	20%	(27/06/2012)
Proyecto:	20%	(25/07/2012)
Evaluación Continua:	15%	
Laboratorio:	10%	
Asistencia:	5%	
Recuperativo/Diferido:		(18/07/2012)

Consultas: CEMISID (40-07) o icaro@ula.ve

1ª Tarea

- Historia de la computadora
 - Evolución de la computadora
 - Generaciones de la computadora
 - Partes de la computadora

Videos de YouTube (Del 1 al 5):

<http://www.youtube.com/watch?v=OJx36OuvaUI>

Condiciones:

- Entregar escrito a mano
- Longitud máxima: 1 Hoja de papel ministro (4 págs)

Fecha de entrega máxima: Miércoles 18/04/2012

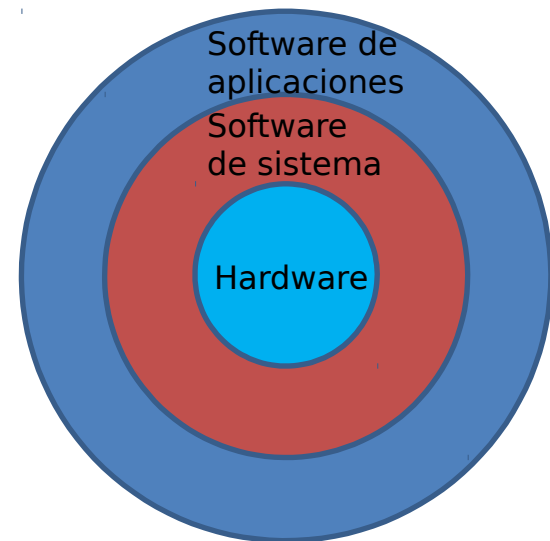
PROGRAMACIÓN 1

Algunas preguntas fundamentales:

- ¿Qué es programar?
- ¿Qué es un programa?
- ¿Qué es un compilador?

Debajo de los Programas

- La electrónica del computador sólo es capaz de ejecutar instrucciones extremadamente sencillas a muy bajo nivel.
- Para llevar una aplicación compleja a estas instrucciones sencillas, se debe pasar antes por varias capas de interpretación y traducción.
- Podemos ver o representar al hardware y software de una forma simplificada como:



- Software de sistema: Sistema Operativo y Compilador (Entre otros).

De lenguaje de alto nivel al lenguaje del hardware

- Para hablar con la electrónica del computador, necesitamos enviarle señales eléctricas.
- En electrónica, resulta conveniente trabajar sólo con dos señales: Voltaje Alto (1 o Cierto) y Voltaje Bajo (0 o Falso).
- Estos dos símbolos o letras, conforman el alfabeto binario, y los denominamos bit (**B**inary **D**igits)
- Los comandos (instrucciones) que le mandamos a las computadoras, son secuencias de bits (Ej: 011001) que indican a la electrónica (el circuito) qué hacer.
- En computación, **programar** es concebir una secuencia de instrucciones binarias (de máquina) para llevar a cabo una tarea o solucionar un problema.

Tipos de Instrucciones

- Los tipos de instrucciones que suele implementar el circuito electrónico de un microprocesador son:
 - Leer un valor almacenado en cierta dirección de memoria
 - Guardar un valor en cierta dirección de memoria
 - Operaciones aritmético - lógicas
 - Evaluar una condición y cambiar de línea en el programa
 - Cambiar de línea en el programa

Lenguaje de máquina y Lenguaje ensamblador

- Los primeros programadores, escribían los programas en lenguaje binario (de máquina); por ejemplo, la siguiente instrucción binaria, podía significar "SUMAR a y b"

1001110101000111

- Después, se inventaron una notación simbólica (El lenguaje ensamblador) que era traducida a lenguaje de máquina a través de un programa denominado Ensamblador.



Lenguaje de alto nivel

- Expresan los algoritmos de una manera adecuada a la capacidad cognitiva humana, no basándose en la capacidad ejecutora de las máquinas.
- No se programa acceso directo al sistema
- No dependen de la arquitectura hardware:
Compilables en múltiples plataformas

Lenguaje de alto nivel

■ Ventajas

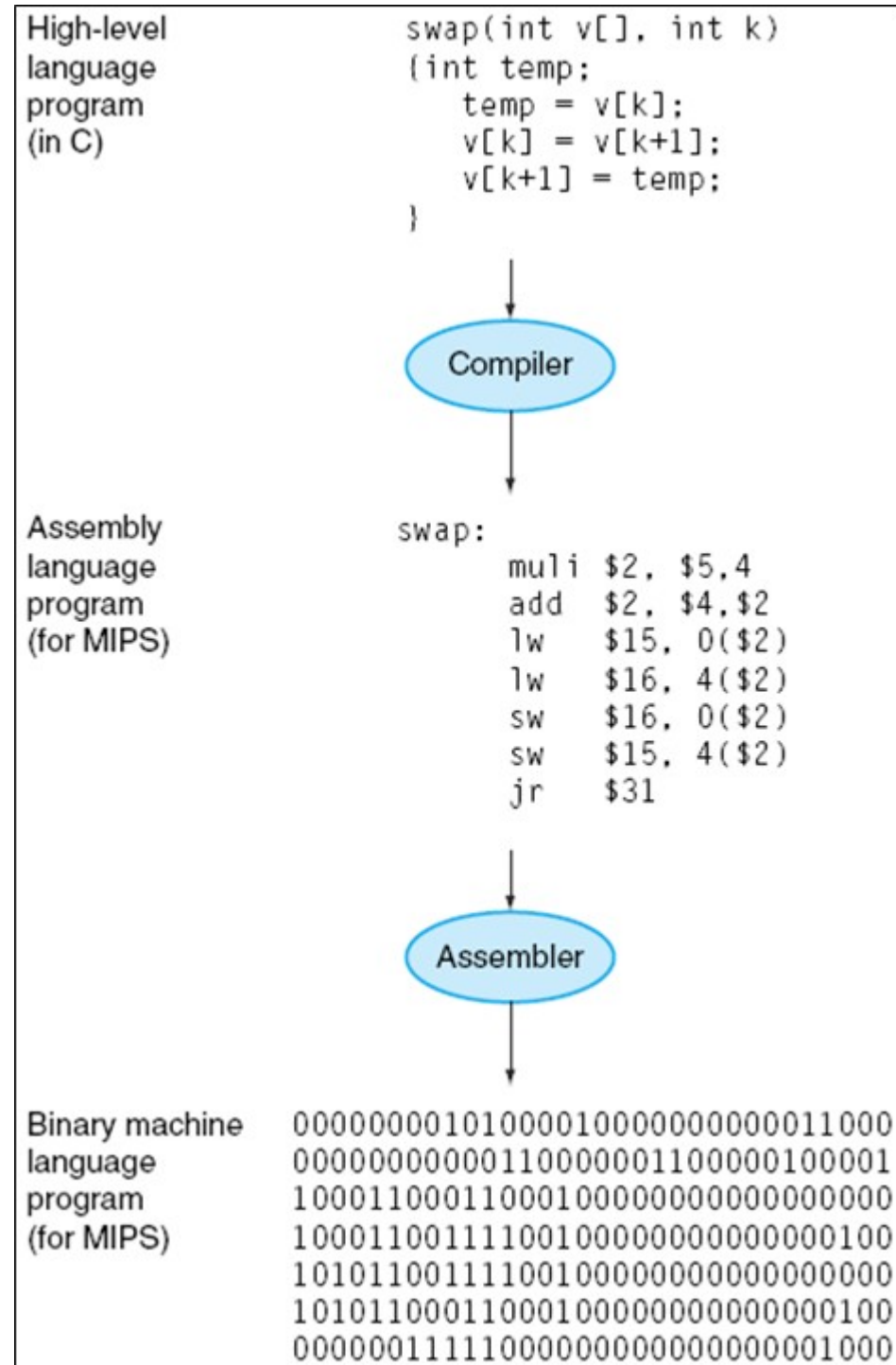
- Se aprende más rápido
- Independiente de la plataforma
- Mejor documentación
- Correcciones más sencillas

■ Desventajas

- Recursos no optimizados
- Se utiliza más tiempo y espacio

Compilador

- Un **compilador** es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación.
- Genera un programa equivalente que la máquina será capaz de interpretar.
- Usualmente el segundo lenguaje es código máquina, pero también puede ser simplemente texto.
- Este proceso de traducción se conoce como **compilación**.



Lenguaje C

- Lenguaje de programación creado en 1972 por Ken Thompson y Dennis M. Ritchie en los Laboratorios Bell
- Evolución del anterior lenguaje B
- Lenguaje de nivel medio
- Compilador: gcc