Microprogramación

Hola!

- Karen Liska

- Ing. En Informática y Sistemas
- Mgtr. Seguridad Informática
- Experiencia en desarrollo de software
- Experiencia en calidad de software
- Karenliska@Gmail.com

1 Objetivos

Objetivos

- Proveer los fundamentos básicos de lenguaje de bajo nivel y la programación de microprocesadores en base a su set de instrucciones.
- Proveer los conocimientos básicos de los lenguajes de bajo nivel y su evolución histórica.
- Conocer la estructura básica de una computadora.
- Que el estudiante conozca de forma muy básica los sistemas numéricos decimal y binario como base de la computación digital.

Objetivos

- Que el estudiante aprenda la división y funcionamiento básico de la memoria principal.
- Que el estudiante cuente con las bases para desarrollar programas en lenguaje ensamblador y base al set de instrucciones de la familia Intel 80x86 y Macroensamblador

2 Distribución de zona

Actividades 📌

- Prácticas de laboratorio: 20
- Proyectos 30

3

¿Por qué lenguaje ensamblador?

Importancia de ASM

- Hoy día, muchas rutinas que son cuellos de botella se traducen en lenguaje ensamblador para minimizar tiempos de ejecución.
- Uso de dispositivos que realizan tareas específicas.
- Crear Virus
- Crear Bootloaders
- Videojuegos

4 ¿Qué veremos?

Lenguaje ensamblador

- Turbo Assembler arquitecturas x86
- Macro Assembler 32bit

5 Entorno de desarrollo

Turboassembler

Herramientas

Virtualización:

- VMWare
- Hyper-V
- VirtualBox
- Parallels

Nota: Deben poder compartir archivos con su máquina virtual.

Herramientas

Sistema operativo

- Cualquier SO Microsoft de **32 bits**

Herramientas

IDE:

- Notepad
- Notepad ++
- VS Code
- Sublime Text


NetBeans
Aca estan los IDEs en los que eh Programado
Programaste en el Block de Notas ?
Ustedes no?

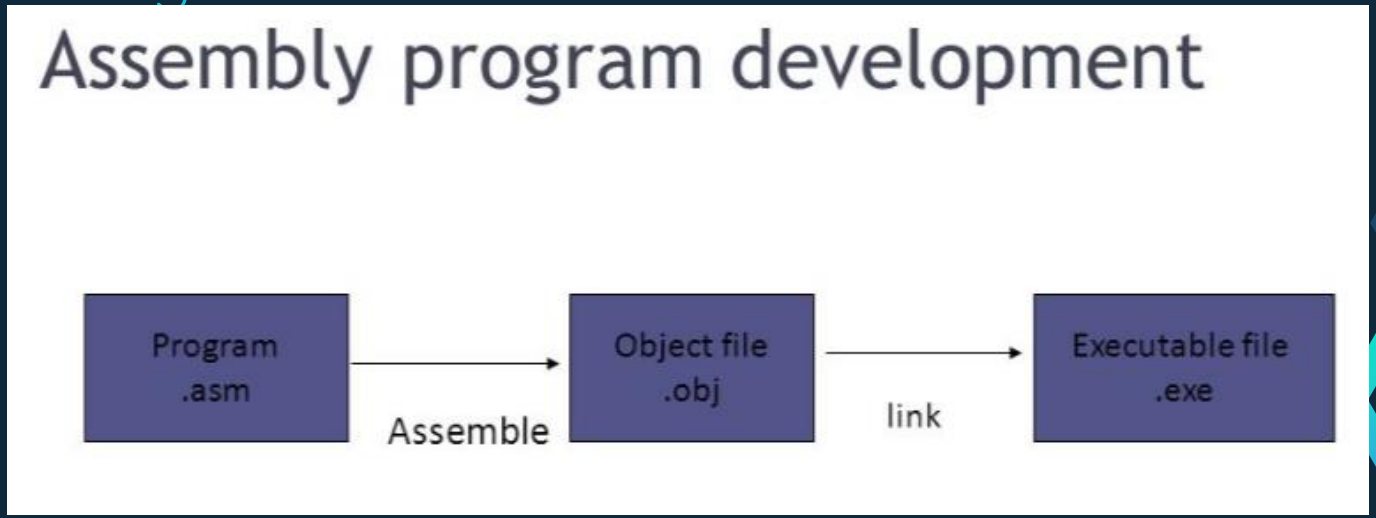
Herramientas

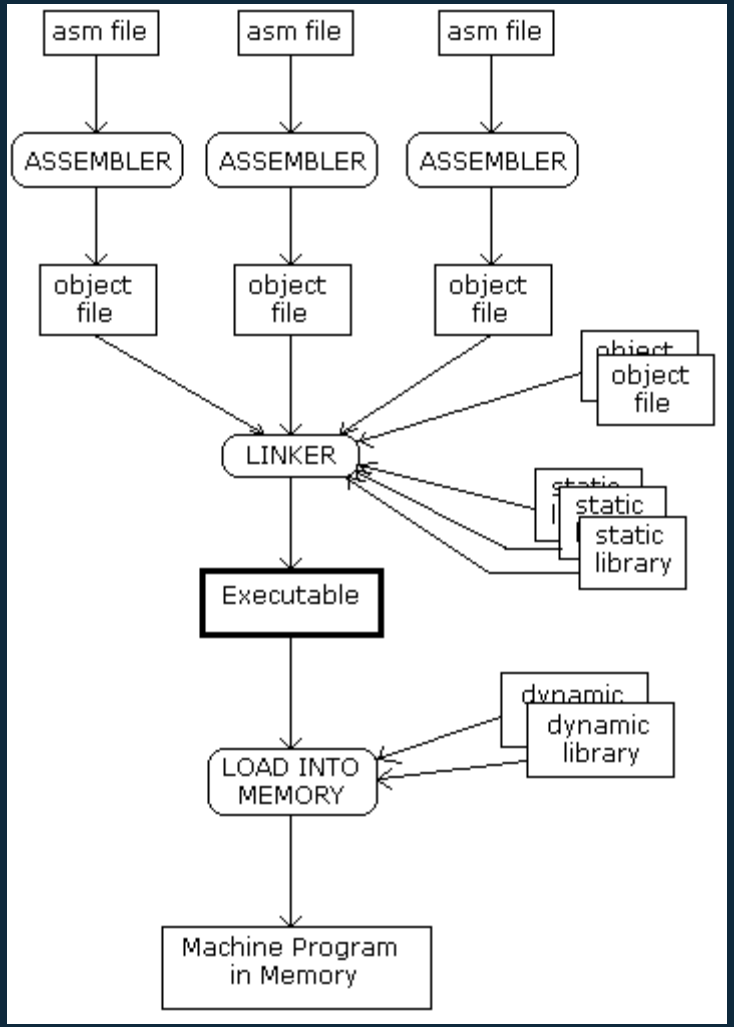
Generación de ejecutable:

- TASM
- TLINK
- CMD

6 Proceso

Flujo para crear ejecutable




asm file
asm file
asm file
ASSEMBLER
ASSEMBLER
ASSEMBLER
object file
object file
object file
object file
LINKER
static library
Executable
dynamic library
LOAD INTO MEMORY
Machine Program in Memory