**Computador** -> Conexiones lógicas -> Programación física a través de un cableado.

El concepto de **software** aparece desde el **almacenamiento de datos**. Los únicos que podían desarrollarlos eran los que desarrollaban **hardware**.

Se atacaba la inteligencia **latinoamericana**. Se buscaba eliminar el conocimiento, y eso hizo que se atrasara el conocimiento unos 10 años.

El **“mundo libre”** comenzó a mejorar el software y avanzo en el mismo. Era mejor pagarle a un **programador** que comprar un hardware más caro. Se necesitaba un **software que funcione en cualquier máquina.** Un ejemplo es el **assembler**, el primer nivel de abstracción, que todas las maquinas entienden.

**Lenguaje C**: 32 términos con los que se puede programar prácticamente cualquier cosa. Se puede manipular memoria y disco rígido. A partir de C salen nuevos lenguajes.

**Lenguajes compilados**: Programa que genera un archivo que luego es ejecutado por un programa.

**Lenguajes interpretados:** Texto que alguien interpreta.

**ROM** de control (RISC – CISC) “Cerebro de la pc”

**RISC**: Solamente utiliza energía dependiendo que se va a hacer.

**CISC**: Necesita mucho procesador para hacer poco o mucho -> Instrucciones más complejas.

**Interpretados**: **Fuentes** -> Están limitadas al intérprete. // **Interprete**: Tiene una serie limitada de instrucciones que puede ejecutar.

**Compilados:** Son más potentes. Más peligrosos. Menos usados. **Fuentes** -> No están limitados. // **Compilador**: El límite que tenemos pasa a ser lo que tenemos programado.

Ambos se ejecutan sobre la maquina mediantes **distintos procesos**. Y todos terminan en assembler -> Traducidos a código **binario** mediante el microprocesador.

**COMANDOS EN LINUX**

**cd** Sirve para entrar al home del usuario (ejemplo: home/alumno)

**cd ..** Sirve para bajar un nivel.

**pwd** Sirve para saber dónde estamos.

**man man** Comando para saber de qué sirve el comando posterior.

**man info** Comando para saber de qué sirve el comando “info”.

**echo “---“ >> “nombre.txt”**  Sirve para crear un archivo, en “---“ va el texto // “nombre.txt” es el nombre de archivo.

**ls** Sirve para listar el contenido de una ubicación.

**cat** Sirve para mostrar contenido (primer nivel).

**more** Sirve para mostrar contenido (segundo nivel).

**less** Sirve para mostrar contenido (tercer nivel).

**tree** Sirve para mostrar contenido en forma de árbol.

**history >> “nombre.txt”** Crea un historial de archivo.

**mkdir** Crea un directorio. Es necesario ser root.

**sudo su –** Sirve para acceder mediante root. Es necesario la contraseña.

**exit** Sirve para salir de root.

~$ home \*\*\*\* /$ root \*\*\*\* ~# root \*\*\*\* /# \*\*\*\*