Unidad 1:

Un sistema operativo explota los recursos hardware de uno o más procesadores para proporcionar un conjunto de servicios a los usuarios del sistema. El sistema operativo también gestiona la memoria secundaria y los dispositivos de E/S (entrada/salida) para sus usuarios. Por tanto, es importante tener algunos conocimientos del hardware del computador subyacente antes de iniciar el estudio de los sistemas operativos.

Registros CPU:

1. \*Registro de datos de E/S\*: Almacena los datos que se intercambian entre la CPU y los dispositivos periféricos. Ejemplo: Registro que guarda los caracteres introducidos desde un teclado a una computadora.

2. \*Registro de dirección de E/S\*: Indica la dirección específica para comunicarse con dispositivos de entrada/salida. Ejemplo: Dirección del controlador del teclado en una computadora.

3. \*Contador de programa (PC)\*: Almacena la dirección de memoria de la próxima instrucción a ejecutar. Ejemplo: La dirección de memoria de la siguiente instrucción en un programa de sumar dos números.

4. \*Registro de instrucción (IR)\*: Temporalmente almacena la instrucción actualmente en ejecución. Ejemplo: La instrucción que indica "sumar los valores en los registros A y B y almacenar el resultado en el registro C".

5. \*Registro de dirección de memoria\*: Almacena la dirección de memoria de los datos a los que la CPU necesita acceder. Ejemplo: Dirección de memoria donde se encuentra el primer número en un programa de suma.

6. \*Registro de datos de memoria\*: Almacena los datos que se leen o escriben en la memoria. Ejemplo: Los números que se cargan en la memoria para realizar la suma en un programa.

- Ejemplo que involucra los 6 registros en acción:

Imagina que estás escribiendo un programa en una computadora para sumar dos números ingresados por el usuario desde el teclado. El programa funciona de la siguiente manera:

1. \*Registro de dirección de E/S\*: La CPU utiliza esta dirección para comunicarse con el controlador del teclado y obtener los números ingresados por el usuario.

2. \*Registro de datos de E/S\*: Aquí se almacenan temporalmente los caracteres introducidos por el usuario desde el teclado.

3. \*Contador de programa (PC)\*: Inicialmente, apunta a la primera instrucción del programa que indica la solicitud de ingreso del primer número desde el teclado.

4. \*Registro de instrucción (IR)\*: Almacena temporalmente la instrucción actualmente en ejecución. Por ejemplo, la instrucción para solicitar el ingreso del primer número desde el teclado.

5. \*Registro de dirección de memoria\*: Almacena la dirección de memoria donde se guardarán los números ingresados por el usuario.

6. \*Registro de datos de memoria\*: Aquí se almacenan temporalmente los números ingresados por el usuario antes de ser procesados por la CPU.

En este ejemplo, cuando el programa solicita el primer número, el controlador del teclado lo envía a través de los registros de datos y dirección de E/S. La CPU utiliza el registro de dirección de memoria para almacenar este número en la memoria y el registro de datos de memoria para retener temporalmente el número mientras se realiza la suma. Luego, el programa solicita el segundo número de manera similar. Finalmente, la CPU utiliza los registros de datos de memoria para acceder a los números ingresados, sumarlos y almacenar el resultado en la memoria utilizando el registro de dirección de memoria. El contador de programa sigue el flujo del programa, apuntando a la siguiente instrucción después de cada paso.

(continuar con el el libro)

Procesos

Un Proceso es una Instancia de un Programa en ejecución.

– Cada ejecución de un programa da lugar a un proceso.

– El proceso -> unidad de procesamiento que gestiona el sistema operativo.

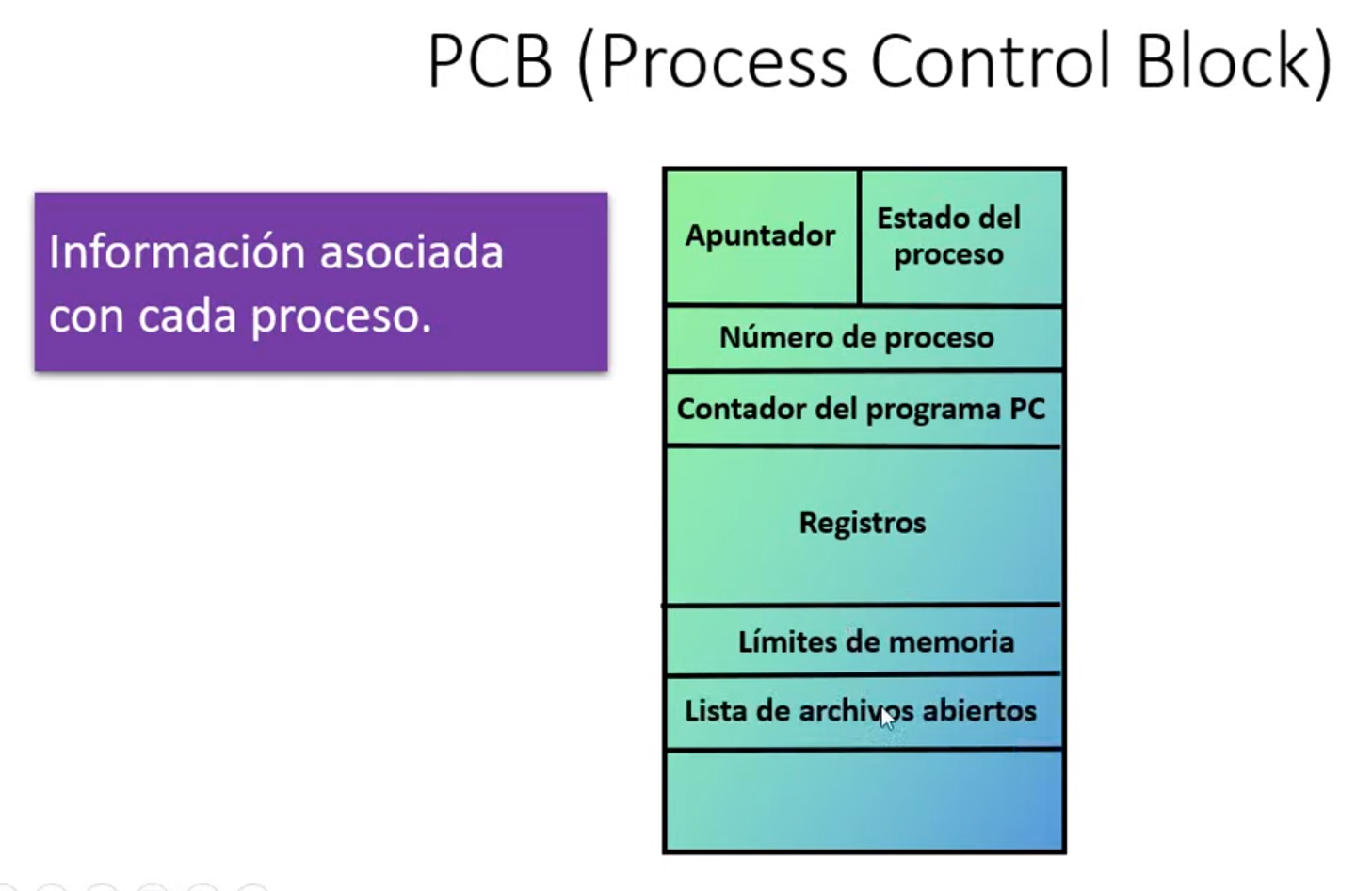
Un proceso está formado por:

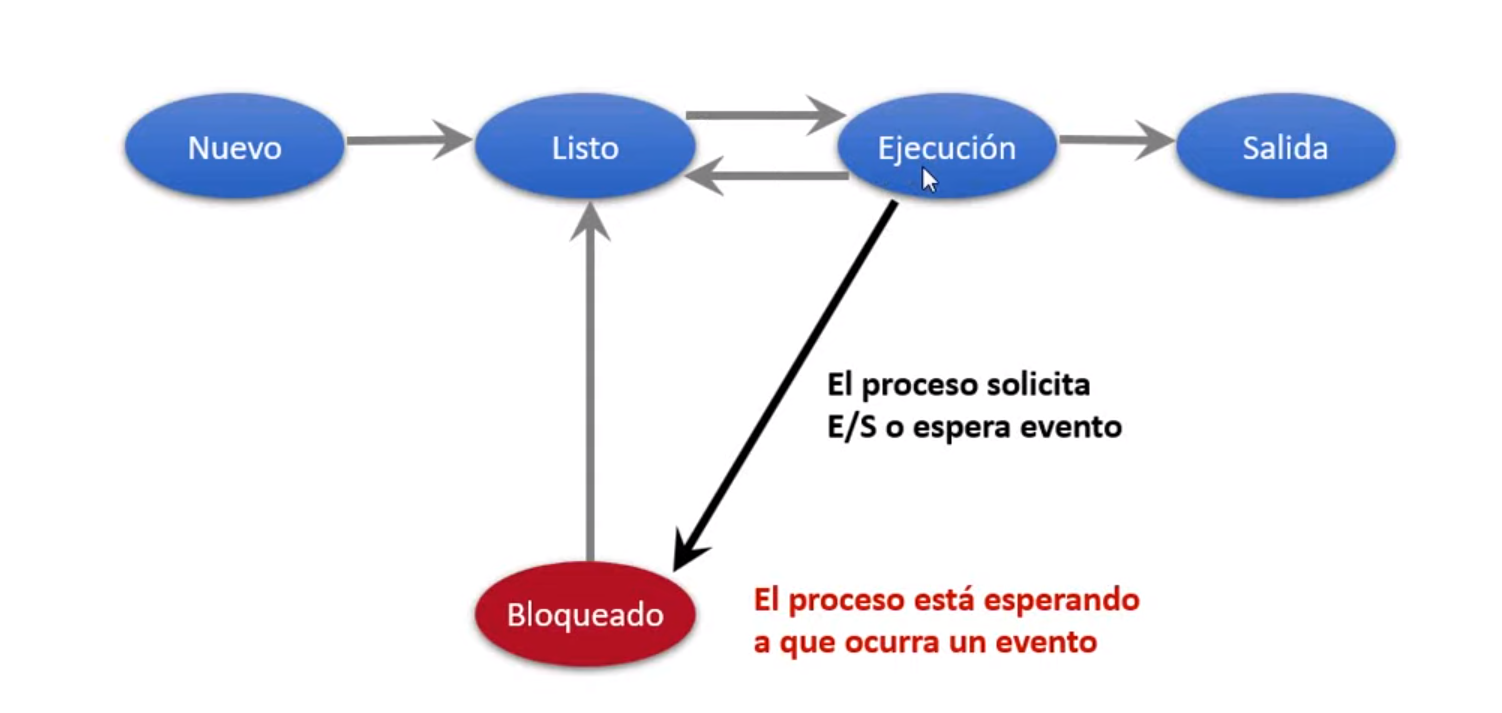
– Código del programa: Instrucciones.

– Conjunto de datos asociados a la ejecución del programa.



El **bloque de control del proceso** o en inglés **PCB** es un registro especial donde el sistema operativo agrupa toda la información que necesita conocer respecto a un proceso particular.





* Entre el estado Listo y Ejecución se lleva a cabo la **Planificación de procesos.**
  + Esta se encarga de asignarle tiempo de CPU a cada proceso:



* El Sistema operativo al cambiar de un proceso A a un proceso B, debe guardar los datos del A en su respectivo PCB.
* Recordar que cuando un proceso A necesita llevar a cabo una E/S manda una llamada al sistema, y el CPU aprovecha y cambia al proceso B.
* Con la llamada al sistema los programas de usuario pasan de modo usuario a modo supervisor (kernel) para que se ejecuten las rutinas del SO. -> Instrucciones privilegiadas (ej.: instrucciones de E/S)
* El CPU tiene bit de modo: 0 para supervisor, 1 para usuario.

**Creación de procesos 🡪** <https://www.youtube.com/watch?v=oIhj2AK8wJE&list=PLJbyYK99gf2X2tkRPsck0Ar8RVVuHnDLp&index=10&ab_channel=Mtro.Jos%C3%A9LuisElvira>

Los SO proveen mecanismos para que los procesos puedan crear otros procesos →Llamada al sistema; El proceso de creación se puede repetir recursivamente creándose una “estructura familiar” → Árbol de procesos, asignando los recursos al nuevo proceso de la siguiente forma:

– Los obtiene directamente del SO

* El padre debe repartir sus recursos con el proceso hijo o compartir todos o parte de ellos con él y así se evita que un proceso bloquee el sistema multiplicándose indefinidamente.

• En términos de ejecución el padre continua ejecutándose en paralelo con su/s hijo/s y el padre espera a que alguno o todos sus hijos hayan terminado

• En términos del espacio en memoria, el proceso hijo es un clon del proceso padre y tiene ya un programa cargado en memoria

**Niveles de planificación**

* Planificación a **corto plazo**: Selecciona que proceso debe ser el próximo en ejecutarse y otorgársele el CPU. Mayoer frecuencia de invocación.
* Planificación a **medio plazo**: Selecciona qué procesos se añaden o se retiran (expulsión a swap) de memoria principal.

• Planificación a **largo plazo**: Selecciona que procesos se llevan a la cola de listos. Menor frecuencia de invocación.

**Tipos de planificación**

• No apropiativa: El proceso en ejecución conserva el uso de la CPU mientras lo desee.

• Apropiativa: El sistema operativo puede expulsar a un proceso de la CPU.

Ejemplos de Planificaciones: (preguntar al profe si va a haber ejercicios de c/u. O si serán ejercicios sobre el tiempo de retorno como los del video con 1 o + CPU. 🡪 <https://www.youtube.com/watch?v=DhahfKvvBBc&list=PLJbyYK99gf2X2tkRPsck0Ar8RVVuHnDLp&index=9&ab_channel=Mtro.Jos%C3%A9LuisElvira> )

* First to Come First to Serve: Primer en llegar primero en servir
* Shortest Job First: Primero el trabajo más corto.
* Cíclico o Round-Robin
* Asignación por prioridades
* Múltiples Colas con retroalimentación.

