**CLASE 8- PROCESOS**

Siempre que queramos que nuestra computadora realice una instrucción son los procesos los que asumen este trabajo. Para que así la CPU ejecute la tarea por medio del plan de acción diseñado por el sistema operativo.   
¿Que son los procesos?  
Cuando la información que tiene ese programa se carga en la memoria y se pone en ejecución se denomina proceso.  
1. Información del programa, 2. Memoria, 3. Procesos= Ejecución de un programa o instrucción.  
Todos los softwares ejecutables se organizan en procesos, que quieren utilizar la CPU y es el SO quien organiza el orden de ejecución de los procesos. A este cambio de proceso lo llamamos cambio de contexto. Los procesos se ejecutan uno a la vez. Estos procesos no se almacenan en la memoria principal ya que consumen espacio y a la larga terminarían por llenar la memoria RAM. Todos los procesos son efímeros. Se crean y se terminan.

Los procesos se crean de diversas formas:  
1. **Proceso de manera interactiva con el usuario**: Por ejemplo, cuando exportamos un archivo estamos generando un proceso.  
2. **Proceso de llamados al SO:** Se crean en segundo plano. Por ejemplo, cuando un software no puede acceder directamente a un recurso entonces le solicita al SO que lo gestione. Para que un proceso termine tiene que pasar por algún estado que determine su condición. Estos estados son:  
\*Nuevo  
\*Listo  
\*Ejecución  
\*Bloqueado  
\*Salida.  
Cuando el proceso se crea se encuentra en el estado **nuevo**, luego pasa a **listo**(preparado) cuando el sistema operativo los carga en la memoria. Cuando se empieza a ejecutar, ingresa al estado de **ejecución**. Aquí hay varios caminos posibles para el proceso. Cambio el contexto, por ejemplo, que sea suspendido cuando hacemos un llamado para leer un archivo(**bloqueado**). Finalmente, cuando el proceso es ejecutado y cumple con su objetivo pasa al estado de **salida**, dando lugar a que otro proceso sea ejecutado. Todo esto sucede en algunos segundos.   
Existen varios mecanismos de comunicación entre procesos o IPC=Inter Process:  
A.Uno de ellos son las señales, las cuales son avisos que pueden guiar un proceso a otro. Luego el SO se encarga que el proceso que recibe la señal tome una acción para gestionarla.  
B. Memoria compartida. Un recurso compartido a disposición de los softwares para que puedan intercambiar información. Entonces dos procesos pueden estar realizándose con la memoria compartida y al mismo tiempo pueden estar intercambiando información.

3. **Hijos:** Cuando se da un proceso que no puede resolverse instantáneamente como, por ejemplo, cuando ocurre una llamada al Sistema. Se van a crear otros procesos que se denominan hijos. Su función es realizar subtareas para lograr que el proceso padre pueda lograr su objetivo. Los procesos padres pueden tener varios procesos hijos, pero los procesos hijos solo pueden tener un proceso padre.

**COMUNICACIÓN DE PROCESOS:**  
Existen dos tipos de procesos que se ejecutan de manera concurrente:   
- Los procesos independientes: Tienen total y completa autonomía, no pueden ser afectados y tampoco afectar a otros procesos que se estén ejecutando en el sistema.  
- Los procesos cooperativos. Si pueden afectar y ser afectados. De hecho, cualquier proceso que comparta cualquier tipo de datos o recursos con otros procesos es considerado cooperativo.   
¿Porque motivo deberíamos hacer una cooperación entre procesos?   
/El primero es que algunos procesos carecen de información asique deben consultarla para poder ejecutarse. Por ello, la información debe ser compartida.  
/El segundo motivo es que gracias a que la información es compartida el CPU trabaja de una manera mas eficiente y veloz. Esto da como resultado la modularidad, es decir, cuando una tarea contiene varios pasos el CPU puede ejecutarlos de manera independiente y simultánea. Modularidad: Ejecución independiente y simultanea de varios pasos de una tarea. Por ejemplo: La línea 1,3,5 pueden estar realizándose en simultaneo pero la 4 necesita que la 2 se haya ejecutado y la 2 necesita que se haya ejecutado la 1.  
¿Puede este tipo de tarea traernos inconvenientes?  
Si, siempre y cuando un proceso ejecute un proceso de forma errónea o no haya planificación. ¿Alguna vez enviaste un archivo a imprimir y surgió un error? Sucede porque el **recurso** se encuentra **bloqueado** por otro proceso que se esta ejecutando. Para que los procesos se comuniquen entre sí, existen **dos métodos** **de intercomunicación** o IPC=Inter-Process Communication. **Que constan de dos modelos**: La memoria compartida y los pasos de mensajes.   
\*En el **modelo de memoria compartida** se establece un espacio en memoria que será compartido por los procesos. Imaginemos que escribimos en una hoja un resumen para un examen y un compañero también necesita estudiar, entonces le prestamos el documento para que lo fotocopie y así no se modifica el nuestro. El a su vez, realiza sus propios apuntes que son fotocopiados por otros compañeros y de esta forma se van compartiendo los recursos entre todos para que cada uno pueda cumplir sus objetivos.  
\* En el caso de **modelos por los pasos de mensajes** imaginemos que nuestro apunte quedo en la fotocopiadora y un alumno que no nos conoce le parece interesante. Entonces le pide al encargado si puede hacer una copia, este a su vez se contacta con nosotros y nos pide el permiso y nosotros accedemos. Aquí el encargado de la fotocopiadora es un **intermediario entre ambos procesos. En los Sistemas Operativos esta tarea la lleva cabo el Kernel.** Ambos tienen ventajas y desventajas de uno con el otro.  
Ventajas:  
-La memoria compartida en general es más económica que utilizar un multiprocesador.  
-En el paso de mensajes no existen los errores como exclusión mutua y son compatibles con cualquier tipo de arquitectura de computadora.

**SINCRONIZACION DE PROCESOS:**Los procesos pueden transitar por diferentes estados, los cuales indican en que parte del ciclo de vida se encuentra el mismo, y en base a este el SO toma decisiones por el. Estados son:  
-Nuevo: Cuando un proceso se crea  
-Listo: Cuando esta en condiciones de ser ejecutado  
-Ejecución: Cuando su turno de utilizar el proceso comenzó.  
-Bloqueado: Cuando esta esperando que un proceso o recurso pueda ser utilizado.  
-Terminado: Cuando ha sido ejecutado y su ciclo de vida finaliza.  
En base a estos estados, existe una herramienta para la sincronización de procesos. Los semáforos. Mientras un proceso se está ejecutando y aparece una llamada de espera, pasa a una lista de bloqueados, y permanece ahí hasta que un proceso diferente le envía la señal de avance y el proceso que permanecía bloqueado se coloca en una fila de espera para utilizar el CPU.  
Existen dos tipos de procesos:  
-Independientes: los cuales trabajan por si mismos sin requerir ninguna ayuda de nadie.  
-Cooperativos: que trabajan en función de los recursos y disponibilidad de otros procesos.  
Área Critica:   
Hay 3 empleados pero solo uno por vez puede usar la caja registradora entonces se genera una fila de espera y utiliza la caja registradora el primer empleado que la utilice. Esto se denomina **Área Critica**. La caja registradora es el procesador, los empleados los procesos y los clientes las operaciones que deben realizar estos procesos.  
Es sumamente importante llevar una buena planificación del uso del CPU, de lo contrario esto lleva a que la cola de los procesos colapse o tenga una **inanición**. Que es que funcione de forma tan deficiente porque le negaría recursos a otros procesos que necesitan ejecutarse, debido a que los recursos necesarios para el procesamiento no están disponibles.   
Para combatir esto existen distintas **técnicas de planificación**, las cuales ayudan a solucionar los grandes problemas de procesos.  
-**FIFO**: First In, Firts Out. Se asigna **tiempo de ejecución** al CPU al primer proceso que lo solicite. Este tipo de planificación es muy parecida a la fila de personas que se hacen cuando queremos comprar en un local. La más conocida.   
-**SJF**: Shortest Job First. La prioridad de ejecución esta dada no por quien llega primero a solicitar el recurso, sino por quien posee el **menor tiempo de ejecución**. Es decir, si dos procesos solicitan al CPU al mismo tiempo primero se procesa el que menor tiempo lleve de ejecución.  
-**SRTF** (un algoritmo parecido al último) Shortest Remaining Time. En el cual, si un proceso largo se está ejecutando y llega un segundo por **tiempo más corto**, se interrumpe el primero ejecutándose el segundo. Una vez terminado este, comienza nuevamente el primer proceso en el sector en el que fue cortado. Al menos que aparezca otro mas chico para terminar su ejecución. **Se interrumpe un proceso por otro que requiera menos tiempo de CPU.**   
-**ROUND ROBIN** : En el cual existe una porción de tiempo (por cantidad de tiempo) establecida o Quantum de tiempo en donde los procesos a medida que van llegando a la fila de espera se ejecutan en el CPU, hasta que el quantum se cumple. Una vez cumplido, se interrumpe el proceso y si es que aun le falta tiempo de ejecución, vuelve a la cola ubicándose al final hasta que es nuevamente su turno y de esta forma se establece que **todos los procesos llevan un tiempo de ejecución equitativo** y todos los procesos serán ejecutados de la misma manera.  
Existen **otras planificaciones** las cuales combinan de forma hibrida las ya explicadas o presentan algoritmos diferentes como ser la **retroalimentación multinivel** o **planificación por comportamientos.** Estableciendo prioridades entre los procesos o cambiando el tiempo de ejecución. Pero como sabemos estos procesos también viven en constante innovación gracias a los avances de las tecnologías.

HILOS DE EJECUCION(THREADS)

“Divide y vencerás”   
Como están compuestos los **procesadores**?   
Físicamente son un conjunto de un gran número de transistores configurados de tal forma que realizan operaciones binarias con impulsos de energía eléctrica. Estos a su vez contienen un núcleo el cual puede ser más de uno. A mayor cantidad de núcleos mayores son los procesos que se pueden ejecutar en paralelo.   
**Procesos:** Un proceso es un conjunto de operaciones que componen a un programa, los cuales a la hora de ejecutarse se reparten la utilización del procesador para realizar su tarea. ¿Pero un proceso, también puede dividirse? Si, un proceso puede dividirse en secuencia de tareas también denominadas **hilos**. Los cuales son porciones de código que pueden **ejecutarse de forma simultánea en cooperación con otros subprocesos**.   
Podríamos verlo como un chef con sus ayudantes en la cocina. Donde cocinar un determinado plato puede repartirse entre sus ayudantes, entonces la eficiencia del proceso al dividir las tareas seria muchísimo mayor. Procesos simultáneos = + Eficiencia.  
**Múltiples hilos pueden existir dentro de un proceso, ejecutándose de forma concurrente compartiendo recursos y memoria.** Los procesos no comparten recursos entre ellos cuando se ejecutan, los hilos sí. Al ser tantos hilos trabajando en conjunto, es muy importante la **sincronización.** Ya que en su proceso puede bloquear un recurso y negarle el acceso a otro hilo.   
Hasta la década de los 2000, los procesadores eran monolíticos, por lo cual solo podían trabajar con un solo hilo a la vez (Monolíticos=un solo núcleo). Luego aparecieron los procesadores multinúcleos (Multinúcleos = varios núcleos), los cuales comenzaron con esta metodología de trabajo de varios hilos de ejecución, aumentando así la velocidad de procesamiento.  
Los sistemas de un solo núcleo tienen:  
- una capacidad de respuesta menor,   
- su comportamiento es más predecible,   
+ no presentan los errores que podrían llegar a presentar los multihilos y   
+ los problemas de bloqueo de recursos bajan considerablemente.  
En comparación, los sistemas multihilos poseen ventajas como:  
+ Excelente capacidad de respuesta a operaciones   
+ Con un buen trabajo en paralelo de sus tareas.  
- La sincronización es compleja de planificar  
- y su comportamiento es difícil de predecir ya que puede presentar errores pasados por alto en la etapa de prueba y desarrollo.

PLANIFICACION DE PROCESOS:  
Son las políticas y mecanismos que poseen los sistemas operativos actuales para realizar la gestión del procesador. Su objetivo es dar un buen servicio a todos los procesos que existan en un momento dado en el sistema. El planificador del procesador tiene como misión la asignación del mismo a los procesos que están en la cola de procesos preparados.