**"02.Administración de Procesos”**

1. **Defina Proceso, Programa y Ejecutable indicando sus diferencias y relaciones.**

*Proceso*: es un programa en ejecución, usa el CPU, y su imagen (PCB, Datos, Instrucciones, Pila) está ubicado en memoria principal (RAM).

*Programa*: es un ejecutable y también un archivo, teniendo un conjunto de instrucciones compiladas que solo puede entender el hardware.

Ejecutable: cuando ejecuto un programa se transforma en un proceso. El programa puede tener varios procesos asociados.

1. **¿Por qué es necesario que el Sistema Operativo administre a los Procesos?**

Es porque estos procesos están usando dos recursos importantes del hardware: CPU y Memoria. Y no todos los procesos usan el CPU en todo momento estando almacenado en memoria principal.

1. **¿Cuáles son las principales tablas utilizadas por el Sistema Operativo para administrar los procesos y recursos de la computadora?**

Las principales tablas son: Tabla de procesos, tabla de E/S, Tabla de Memoria, Tabla de Archivos.

1. **¿Qué información de un Proceso almacena en memoria el Sistema Operativo? Indique los atributos principales y su finalidad. (buscar pag.137 Stallings)**

La información almacenada en memoria para un proceso es la imagen del proceso. Y son:

Identificador: id proceso, id usuario, información para identificar un proceso que el sistema operativo se lo asigna automáticamente.

Información de estado: se va a guardar todos los registros del proceso cada vez que hace un cambio de los Estados.

Información de control: tiene el estado del proceso, la información de que porción de memoria tiene asignado, que recursos de E/S esta usando. Toda información que el SO necesita para administrar al proceso/s.

1. **¿Qué es el Bloque de Control de Procesos (en inglés Process Control Block o PCB)? ¿Para qué se utiliza?**

La PCB lo que tiene es toda la información que necesita el sistema operativo para controlar y administrar el proceso. Y se utiliza para tener el identificador del proceso, la información de Estado del procesador, y la información de Control del proceso.

1. **¿Cuáles son los posibles estados de un Proceso? Indique el diagrama de interacción que utiliza 3 y 5 estados, indicando sus diferencias.**
2. **¿Cuáles son los principales motivos para crear un nuevo proceso? ¿Quién se ocupa de realizarlo?**

1.Nuevo proceso de lotes:

2.Sesión interactivo

3.Creado por el SO para proporcionar un servicio

4.Creado por un proceso existente

1. **¿Cuáles son los principales motivos para terminar un proceso existente? ¿Quién se ocupa de realizarlo?**
2. **¿Cuáles son los principales motivos para cambiar el proceso que se está ejecutando? ¿Quién se ocupa de realizarlo?**
3. **¿Por qué es conveniente distinguir a los procesos que se están ejecutando, los que están “listos” para ser ejecutados y los que están realizando una operación de entrada/salida?**
4. **¿Cuál es la ventaja de “suspender” procesos? ¿Quién la realiza?**

El “S.O.” puede tener procesos funcionando y que no esté en la memoria principal. Ese estado es el Estado de “Suspendido”. Un proceso que está en estado suspendido es un proceso que toda su imagen se guarda en el disco rígido, la ventaja de suspender es “Liberar Memoria”.

1. **¿Qué es un Hilo (o Thread)? ¿Cuál es su diferencia y relación con los Procesos?**

Cada Hilo ejecuta ciertas instrucciones del proceso, siendo mas barato para crear. Porque no se necesita crear toda una PCB nueva, todos los datos nueva, toda la pila nueva.

La diferencia es en vez de hacer que un proceso genere muchos hijos, lo que se puede hacer es dividir en varios hilos. Buscar en la 183

1. **¿Qué significa que un Sistema Operativo es multi-hilo (en inglés multithreading)? ¿Qué ventajas tiene?**
2. **¿Qué información almacena un Sistema Operativo multi-hilo sobre los Hilos?**
3. **¿Cuáles son los principales estados de un hilo?**

Los estados de los hilos, al igual que con los procesos son: Ejecutando, Listo y Bloqueado

1. **¿Se pueden utilizar Hilos si el Sistema Operativo no es multi-hilo? ¿Cómo? ¿Qué ventajas y desventajas posee?**