

PROYECTO COMPILANDO CONOCIMIENTO

PROGRAMACIÓN

Bases de Sistemas Operativos

Una Pequeña (Gran) Introducción

AUTORES:

Rosas Hernandez Oscar Andrés

Lopez Manriquez Angel

Índice general

I	Una Introducción	2
1.	Introducción	3
1.1.	¿Qué es un Sistema Operativo?	4
1.1.1.	Definición Formal	4
1.2.	Tipo de Sistemas	5
1.3.	Terminos Básicos	6
2.	Partes del Sistema Operativo	8
2.1.	Vista General	9
2.2.	Capas	10
2.3.	Interfaces	11
3.	Funcionamiento de un Procesador	12
3.1.	Pipeline	13
4.	Kernel de un Sistema Operativo	14
4.1.	Conozcamos a los Administradores	15
5.	Administrador de Procesos	17
5.1.	Introducción a Procesos	18
5.2.	Bloque de Control de Proceso (PCB)	19
5.3.	Árbol de Procesos	21
5.4.	Procesos en C	22
5.4.1.	Sustitución de Código	22

Parte I

Una Introducción

Capítulo 1

Introducción

1.1. ¿Qué es un Sistema Operativo?

Los sistemas operativos surgen como una solución a la problemática de la **administración de un equipo de computo**, de forma tal que fuese más sencillo trabajar con el y aprovechar al máximo sus recursos.

Un sistema operativo es un programa encargado de controlar todos los recursos de una computadora"

1.1.1. Definición Formal

“Un sistema operativo es un software de base compuesto por un conjunto de administradores encargados de la administración de cada uno de los recursos de un equipo de computo de forma rápida y eficiente.”

Características

- Un sistema operativo es un software de base debido a que es una plataforma que permite la creación y ejecución de aplicaciones desarrolladas para el propio sistema. Como software de base, el sistema operativo ofrece interfaces para la creación o ejecución de las aplicaciones desarrolladas.
- Un sistema operativo está compuesto de un conjunto de administradores, los cuales controlan todos los recursos del equipo de computo, estos administradores son:
 - Administrador de Procesos
 - Admin. de Memoria
 - Admin. de Entrada / Salida
 - Admin. de Archivos
 - Admin. de Red
- Un sistema operativo debe ejecutarse lo más rápido posible, evitando quitarle tiempo de procesamiento a las aplicaciones de los usuarios, por otro lado debe administrar cada uno de los recursos del equipo de cómputo de forma eficiente, maximizando el uso de cada recurso controlado. La rapidez y eficiencia es uno de los principales u objetivos que un Sistema Operativo debe cumplir durante su ejecución.

1.2. Tipo de Sistemas

Sistemas Genericos:

- Por lotes sencillos
- Por lotes multiprogramados
- De tiempo compartido

Sistemas Distribuidos:

- Distribuidos: En estos los procesadores no comparten el mismo reloj ni memoria, son sistemas debilmente acomplados
- Paralela: Los procesadores comparten los recursos como la memoria, son sistemas fuertemente acomplados

1.3. Terminos Básicos

- **Spooling:**

SPOOL significa “Simultaneous Peripheral Operation On-Line”.

En realidad, lo que sucede aquí es que hay buffer para almacenar los datos, por lo general el disco.

Los dispositivos de entrada / salida no pueden trabajar a la velocidad de una CPU. Por lo tanto, la salida de la CPU se almacenará en este spool (buffer) y los dispositivos de entrada / salida pueden tomar la salida de este buffer como y cuando se requiera de acuerdo a su velocidad.

Por lo tanto, la CPU no está vinculada a este dispositivo de entrada / salida y puede realizar otras operaciones. Gracias al spooling podemos mantener la CPU y los dispositivos de entrada y salida trabajando a altas velocidades sin esperar el uno al otro.

- **Reserva de Trabajo:**

Conjunto de trabajos en el disco listos para ser ejecutados por la CPU

- **Planificación de Trabajo:**

Es la técnica que se encarga de seleccionar cual será el siguiente trabajo ejecutado en la CPU.

- **Multiprogramación:**

Técnica utilizada para almacenar múltiples trabajos simultáneamente en la memoria física (RAM).

- **Tiempo Compartido:**

Técnica utilizada para asignar un tiempo de ejecución a cada proceso lo suficientemente corto para conmutar entre ellos.

El sistema de tiempo compartido es donde cada proceso se asigna un período de tiempo determinado y el proceso tiene que terminar su finalización dentro de ese lapso de tiempo.

Si no se logra completar su ejecución, entonces el control de CPU pasa al próximo proceso.

- **Concurrencia:**

Técnica utilizada ejecutar múltiples trabajos bajo la apariencia de simultaneidad o paralelismo mediante una ejecución secuencial.

- **Memoria Virtual:**

Técnica utilizada para aumentar o extender la memoria física (RAM) mediante el uso de una pequeña región de disco.

- **Sistema de Archivos:**

Estructura de almacenamiento de información mediante entes llamados archivos y directorios.

- **Sistemas Paralelos:**

Sistemas utilizados para el multiprocesamiento compuesto por un conjunto de procesadores que comparten el reloj, memoria y buses del equipo, por lo que se conocen como fuertemente acoplados.

- **Sistemas Distribuidos:**

Sistemas utilizados para el multiprocesamiento compuesto por un conjunto de sistemas de cómputo completo que manejan de forma independiente cosas como el reloj, la memoria o los buses. Por esto se le conoce como debilmente acoplados.

Capítulo 2

Partes del Sistema Operativo

2.1. Vista General

Un Sistema Operativo como cualquier otro software sigue un modelo de ingeniería de software para su diseño y construcción, en modelo que se usa es el denominado por capas, este modelo nos dice que cada capa se encarga de realizar una funcionalidad concreta dentro del sistema operativo.

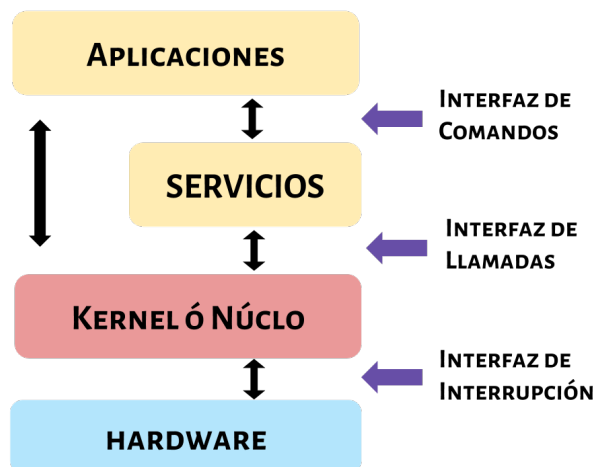
Un sistema operativo normalmente está integrado por las siguientes capas:

- Hardware
- Kernel
- Servicios
- Aplicaciones

Estas capas para llevar a cabo sus funciones requieren comunicarse con cada adyacentes, para lograr esta comunicación se usan las interfaces, estas son:

- Interfaz de Comandos
- Interfaz de Llamadas al Sistema
- Interfaz de Interrupciones

Graficamente las podemos ver como:



2.2. Capas

- **Capa de Aplicaciones**

Esta capa se encarga de mantener cualquier aplicación que el usuario ejecutará en el sistema operativo, siendo la capa con la que el usuario tendrá contacto.

Se encarga de mantener todas las aplicaciones que tu conoces normalmente.

- **Capa de Servicios**

Esta capa se encarga de mantener los servicios que apoyan el funcionamiento del sistema operativo, teniendo servicios de seguridad, de mantenimiento, entre otras.

A veces se suele unir y dividir sus funciones entre la capa de aplicaciones y el kernel.

- **Capa de Kernel o Núcleo**

Esta es la capa principal del sistema operativo, esta mantiene a los 5 administradores que componen a todo sistema operativo.

- **Capa de Hardware**

Esta es la capa mas baja del sistema, se encarga de mantener todas las interfaces de comunicación con el hardware.

2.3. Interfaces

■ Interfaz de Comandos

Esta interfaz comunica a la capa de aplicaciones con la de servicios o a la capa del kernel.

Esta compuesta de por todos los comandos disponibles en el sistema operativo, siendo la interfaz de interacción inmediata que el usuario posee para comunicarse con el sistema operativo.

■ Interfaz de Llamadas al Sistema

Esta interfaz comunica a la capa de servicios con la de aplicaciones o a la capa del kernel.

Esta compuesta por unas APIs (es decir funciones o métodos) que el sistema operativo pone a disposición de los usuarios a través de un lenguaje de alto nivel, por esto se le conoce como una comunicación indirecta.

■ Interfaz de Interrupciones

Esta interfaz comunica a la capa del kernel con la del hardware.

Esta compuesta por un conjunto de interrupciones o servicios de interrupción que el sistema operativo pone a disposición de los usuarios por medio de un lenguaje de programación de bajo nivel, por esto se le conoce como una comunicación indirecta.

Capítulo 3

Funcionamiento de un Procesador

3.1. Pipeline

Pipelining es una técnica de implementación donde múltiples instrucciones se superponen en la ejecución.

La tubería de la computadora se divide en etapas. Cada etapa completa una parte de una instrucción en paralelo. Las etapas se conectan una a la otra para formar un tubo - las instrucciones entran en un extremo, avanzan por las etapas y salen al otro extremo.

Pipelining no disminuye el tiempo para la ejecución individual de la instrucción. En su lugar, aumenta el rendimiento de la instrucción. El rendimiento de la tubería de instrucciones está determinado por la frecuencia con que una instrucción sale de la tubería.

Debido a que las etapas del tubo están enganchadas juntas, todas las etapas deben estar listas para proceder al mismo tiempo. Llamamos el tiempo requerido para mover una instrucción un paso más en la tubería un ciclo de máquina. La longitud del ciclo de la máquina está determinada por el tiempo requerido para la etapa de tubería más lenta.

El objetivo del diseñador de la tubería es equilibrar la longitud de cada etapa de la tubería.

Capítulo 4

Kernel de un Sistema Operativo

4.1. Conozcamos a los Administradores

Podemos separar el Kernel en varios administradores, todos interactúan entre sí para realizar correctamente su trabajo.

- **Administrador de Procesos:**

Se encarga de la ejecución de cualquier trabajo en el Sistema, está compuesto por un conjunto de algoritmos de planificación y de estructuras de datos.

El hardware con el cual interactúa este administrador es el procesador del equipo de cómputo.

- **Administrador de Memoria:**

Este administrador se encarga de la gestión tanto de la memoria física como de la memoria virtual.

Está compuesto por un esquema global de gestión de memoria, así como el conjunto de algoritmos para el control de la memoria.

- **Administrador de Entrada y Salida:**

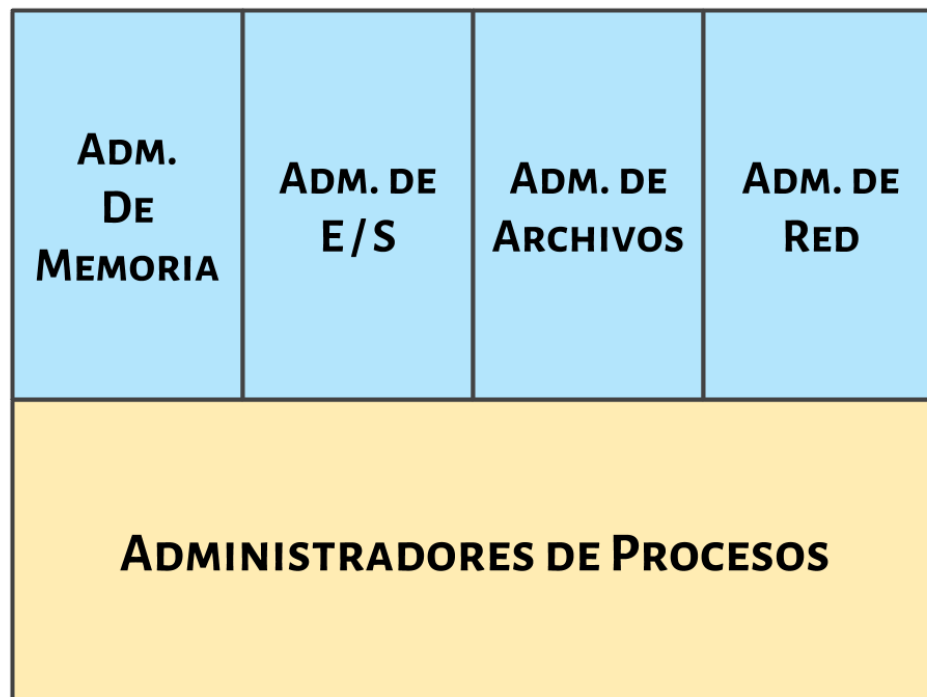
Este administrador se encarga de la gestión de acceder a cualquier dispositivo externo. Está compuesto por un conjunto de interfaces de conectividad y controladores de nivel de hardware como software.

- **Administrador de Archivos:**

Este administrador se encarga de la gestión de todos los archivos del sistema operativo. Se encarga de la organización lógica de la información.

- **Administrador de Red:**

Este administrador se encarga de cualquier comunicación en red del sistema operativo, está compuesto por un modelo de comunicación en red y diversos protocolos asociados al modelo usado.



Capítulo 5

Administrador de Procesos

5.1. Introducción a Procesos

Es uno de los principales administradores del sistema operativo. Este se encarga de la gestión de cualquier trabajo a ejecutar en el sistema. Los trabajos que son ejecutados por este administrador son llamados **procesos**. Se encarga de llevar a cabo la ejecución de todos los programas del sistema operativo.

“Un proceso es un programa en ejecución”

“Un proceso es la entidad mínima de software ejecutándose en el Sistema Operativo contenido en una estructura de información”

De esta definición podemos obtener dos características muy importantes:

- Es una Entidad de Software

Esto implica que tendrá asociado un ciclo de vida. Este ciclo representará las diversas etapas por las cuales puede transitar un proceso.

Un proceso en el sistema operativo se implementará a través de una estructura de información, la cual contendrá toda la información necesaria para caracterizar a dicho proceso.

Esta estructura de información es implementada mediante una estructura de datos en el lenguaje de alto nivel que estemos utilizando.

A esta estructura se le conoce como **Bloque de Control del Proceso** (PCB).

Este PCB es el elemento con el que el Administrador de Procesos interactuará durante el ciclo de vida del proceso presente en el sistema operativo.

- Representa una estructura de información

5.2. Bloque de Control de Proceso (PCB)

La implementación de un proceso en un sistema operativo se lleva a cabo mediante una estructura de datos, esta es conocida como un PCB. Este mantiene toda la información importante de un proceso, esta nos permite caracterizar un proceso en el sistema operativo.

Cada sistema implementa estas estructuras de manera diferente, pero en general todas contienen un:

- **Apuntador:**

Mantiene una referencia del PCB en la cola de planificación

- **Identificador:**

Es un simple entero positivo que se encarga de identificar al proceso

- **Contador de Programa:**

Mantiene el valor del registro del procesador PC ó IP, es decir, la dirección de la siguiente instrucción a ejecutar por el proceso

- **Registros:**

Estos datos mantienen los valores de los registros del procesador que actualmente tiene el proceso en ejecución.

- **Estado del Proceso:**

Este dato mantiene el estado del ciclo de vida actual, en el que se encuentra la ejecución de un proceso.

- **Información de Planificación:**

Estos datos mantienen información relacionada a la forma en que un proceso será planificado, entre esta información se encuentra la siguiente:

- Algoritmo de planificación utilizado
- Prioridad del proceso (en caso de ser utilizada)

- **Información de Memoria:**

Estos datos mantienen información relacionada con la administración de memoria usada para la ejecución del proceso, esta información incluye lo siguiente:

- Páginas o segmentos usados por el proceso
- Tabla de páginas o segmentos utilizados por el proceso

■ **Información Contable:**

Estos datos mantienen informacion relacionada con la administración de memoria usada para la ejecución del progcso, esta información incluye lo siguiente:

- Páginas o segmentos usados por el proceso
- Tabla de páginas o segmentos utilizados por el proceso

■ **Información de Memoria:**

Estos datos mantienen información relacionada con diferentes tiempos consmidos por el proceso, entre estos tiempos se encuentran las siguientes:

- Tiempo de acceso a memoria
- Tiempo de acceso a dispositivos de entrada y salida.
- Tiempo de uso de la CPU

El administrador de procesos basa su operación fundamentalmente en la manipulación de los PCB de los procesos actualmente en ejecución en el sistema operativo, siendo su unidad básica funcional.

5.3. Árbol de Procesos

Internamente, un administrador de procesos utiliza una estructura no lineal para organizar todos los procesos actualmente en ejecución, esta estructura no lineal es conocida como árbol de procesos.

Un árbol de procesos esta formado por un nodo raíz conocido como proceso principal, cada sistema operativo nombra a este proceso principal de forma particular, sin embargo, la función de este proceso es la misma, funciona como la base para la creación de cualquier proceso en el sistema operativo.

Gráficamente un árbol de procesos básico se muestra a continuación:

El árbol de procesos clasifica los procesos en dos tipos:

- **Proceso de Sistema**

Los procesos de sistema son aquellos que conforman al sistema operativo, se caracterizan por ejecutarse en un modo kernel del procesador conocido como modo kernel. En el modo kernel de un procesador se tiene permitido el acceso sin algún tipo de restricción a cualquier recurso del procesador y del equipo de cómputo.

- **Proceso de Usuario**

Los procesos de usuario son todos aquellos procesos ejecutados por los usuarios del sistema operativo, se caracterizan por ejecutarse en un modo del procesador conocido como modo usuario.

En el modo usuario de un procesador se restringe el acceso a recursos críticos tanto del procesador como del equipo de cómputo.

En el Árbol de Procesos se le asigna un identificador para referenciar a los procesos, este identificador es un valor entero positivo único, que se asigna una vez que se crea el proceso. Cualquier referencia se lleva a cabo a través de este identificador.

Dentro de este Árbol se establece una relación Hijo-Padre entre los procesos, esta relación permite heredar características de un proceso padre a sus hijos, además nos permite mantener una organización centralizada.

Esto nos dice que no existen procesos aislados, todo proceso tiene que estar contenido en el Árbol de Procesos.

5.4. Procesos en C

5.4.1. Sustitución de Código

La creación de procesos por sustitución de código es la forma mas común de crear nuevos procesos, todos los sistemas operativos cuentan con esta opción y tiene dos características muy importantes:

- El nuevo proceso sustituye y destruye al proceso padre
- La memoria utilizada por el proceso creador es reutilizada por el nuevo proceso

Los sistemas operativos ofrecen alternativas para que el nuevo proceso no destruya al proceso padre, permitiendo que esta practica sea útil en la vida real.

Por ejemplo UNIX y GNU/Linux combinan tanto la creación por copia de código como por sustitución para permitir que tanto el proceso creador como el nuevo proceso estén en ejecución concurrente.

Por otra parte Windows asocia un hilo a cada proceso que ejecuta y este hilo se ejecuta por sustitución de código, conservando al proceso padre.

Hilos

Son procesos ligeros, es una alternativa a las aplicaciones concurrentes.

Un hilo es un componente de un proceso tradicional que se ejecuta en concurrencia con sus procesos creador.

Para el sistema operativo es más fácil crear hilos que crear un proceso tradicional, por eso se le llaman procesos ligeros.

- Comparten valores de los recursos asignados a su proceso creador, siendo la memoria el principal recurso compartido
- Manejan de manera independiente tanto al pila como el contador de programa
- Dependen de la ejecución de su proceso padre

Los sistemas operativos implementan sus hilos a través de dos posibles métodos:

- Por una biblioteca a nivel usuario: Esta forma es más portable, pues basta con que el sistema computacional soporte la librería para que la app funcione como si nada, pero por la capa extra que le ponemos suelen ser más lentas
- Por llamadas al sistema mediante Kernel: Es lo inverso, perdemos portabilidad, pero por otro lado ganamos velocidad