

Centro Universitario de Ciencias Exactas e Ingenierías.
Departamento de Ciencias Computacionales.



Materia:

Sistemas Operativos.

Profesora:

Becerra Velázquez Violeta del Rocío.

Alumno:

Maldonado Melendez Diego Alberto.

Código:

221977845.

Carrera:

Ingeniería en Computación.

Sección:

D04

Título de la investigación:

Introducción a los Sistemas Operativos.

Fecha:

27 de agosto de 2023.

Índice

Tabla de Imágenes	3
1. Evolución de los Sistemas Operativos.	4
1.1 Procesamiento en serie.....	4
1.2 Sistema de procesamiento por lotes.....	5
1.3 Sistemas de multiprogramación de lotes.....	6
1.4 Tiempo Contemporáneo.....	6
2. Sistemas Operativos y sus objetivos.....	7
2.1 Estructura del Sistema Operativo.....	8
2.2 Funciones del Sistema Operativo.....	9
2.3 Modos del Sistema.....	10
2.4 Tipos de Sistemas Operativos.....	11
2.4.1 Sistemas Operativos por su estructura.....	11
2.4.2 Sistemas Operativos por modos de explotación.....	11
2.4.3 Sistemas Operativos por los servicios ofrecidos.....	12
2.4.4 Por la manera de ofrecer los servicios.....	12
3. Preguntas.....	13
3.1 Temas en común en los libros buscados.....	13
3.2 ¿Qué es un archivo por lotes?	15
3.3 Explicación de procesamiento en serie.....	15
3.4 Definición de Sistema Operativo.....	15
3.5 Lista de los gestores del Sistema Operativo.....	15
3.6 Objetivos de un Sistema Operativo.....	16
3.7 Ilustración y explicación del ciclo Fetch.....	17
3.8 Clasificación de los sistemas operativos.....	17
4. Archivo Por lote.....	18
5. Conclusiones.....	20
6. Referencias.....	21

Tabla de Imágenes.

Figura 1. IBM 1401 Processing Unit.	5
Figura 2. IBM 7094 Consola.	5
Figura 3. Ciclo Básico de una Instrucción.	7
Figura 4. Bibliografía empleada para la investigación.	14
Figura 5. Ilustración Ciclo Fetch junto con Ciclo de Ejecución.	17
Figura 6. Codificación del archivo por lote con 10 instrucciones.	18
Figura 7. Muestra del funcionamiento del archivo por lotes.	19

1. Evolución de los Sistemas Operativos.

El matemático inglés Charles Babbage (1792-1871) fue el diseñador de la primera computadora digital real. Pese a sus esfuerzos y recursos invertidos en construir su "máquina analítica", Babbage nunca hizo que funcionara correctamente porque era una máquina mecánica y la tecnología de la época no podía producir las partes necesarias con la precisión requerida. Por lo tanto, la máquina analítica de Babbage no contaba con un sistema operativo. Hasta los 40's se conoció como la Generación 0, ya que las máquinas no tenían sistema operativo y era necesario conocer el lenguaje binario.

1.1 Procesamiento en serie.

Babbage se dio cuenta que era necesario un software, pero no fue hasta mitades de la década de 1940 cuando John Atanasoff y Clifford Berry construyeron una computadora digital funcional, la cual requería de 300 tubos de vacío para su correcto funcionamiento. Esta época, llamada Primera Generación, fue de avances exponenciales en el área del cómputo, aunque ahora sabemos que tardaban mucho tiempo en realizar cálculos, este fue el preámbulo a lo que hoy conocemos. Se empezó a ahondar en programas de bit a bit, utilizando interruptores mecánicos. Era comunicación directa entre el programa y los componentes de la máquina.

La manera en la que se ejecutaba dicha comunicación era mediante una consola que consistía en varias luces conectadas, switches, especie de dispositivos de entrada, y una impresora en la que se plasmaban los resultados. Mediante los dispositivos de entrada eran ingresados los programas en lenguaje máquina. Era un procesamiento serial porque era de este modo como los usuarios tenían acceso a los dispositivos. Este tipo de sistemas presentaba problemas en la planificación de tareas. Para usar el ordenador, había que apuntarse en una hoja de papel. A veces, alguien terminaba antes de lo previsto y dejaba el ordenador sin usar. Otras veces, alguien se quedaba sin tiempo y tenía que dejar el ordenador sin resolver el problema.

Para ejecutar un programa, llamado trabajo, había que cargar en la memoria el compilador y el programa en lenguaje de alto nivel (programa fuente), guardar el programa compilado (programa objeto) y luego cargar y enlazar el programa objeto y las funciones comunes. Cada paso implicaba montar, desmontar cintas o preparar pilas de tarjetas. Si se producía un error, el desafortunado usuario solía tener que volver al principio de la secuencia de preparación. Así, se perdía mucho tiempo solo en preparar el programa para que funcionara.

La segunda generación, que se dice va desde 1955 hasta 1965, se introducen los transistores, lo que hizo máquinas mucho más estables y eficientes. Ahora el proceso de ejecución era diferente, para ejecutar un programa o conjunto de programas primero era necesario realizar la escritura en papel, para posteriormente pasarlo a tarjetas perforadas y ser entregado a algún operador, que estaba en el cuarto de entradas, para pasarlo al dispositivo y posteriormente recoger los resultados en el cuarto de salidas.

1.2 Sistema de procesamiento por lotes.

Fue también en la segunda generación que surgió el concepto de sistema operativo, y el sistema de procesamiento por lotes o batch fue de vital importancia para esto. Se dice batch OS es el primer sistema operativo usado, desarrollado por General Motors para usarlo en una IBM 701. Retomando un poco los ideales de Babbage, ellos querían que existiese la posibilidad de controlar toda la máquina mediante lo que hoy conocemos como monitor. Así surgen los conceptos de tarjetas y de operador, que es el encargado de introducir todo el lote de trabajo en el área de entrada de la computadora, para retornar el resultado en el monitor.

El propósito de este sistema era juntar los lotes de trabajo en la sala de entrada de datos y luego copiarlos a una cinta magnética usando una computadora, como la IBM 1401, que era muy buena para leer las tarjetas, copiar cintas e imprimir los resultados, pero no tan buena para los cálculos numéricos. Para hacer los cálculos numéricos se usaron otras máquinas mucho más caras, como la IBM 7094.



Figura 1. IBM 1401 Processing Unit.



Figura 2. IBM 7094 Consola.

En síntesis, todo inicia con una tarjeta \$JOB, donde se le indica el tiempo máximo que se debe tardar la ejecución en minutos, el número de cuenta al que se va a cargar y el nombre del programador. Posteriormente se requiere de una tarjeta \$FORTRAN, indicativo que se debe llamar al compilador de igual nombre, para luego llamar a una tarjeta \$LOAD, que notifica al sistema operativo que debe cargar el programa recién compilado. La siguiente tarjeta es la \$RUN, que es la que da continuidad al programa con los datos sucesores, para finalmente usar la tarjeta \$END, que marca el fin del trabajo.

Los lotes se enfilan y se procuraba realizar los trabajos lo más pronto fuese posible. Fue así como surgió una versión primitiva de un lenguaje de control de trabajo (JCL). Lo destacado del sistema de procesamiento por lotes era su capacidad de organización de los trabajos. También es cierto que tal avance en la ejecución de programas requería una mejoría en aspectos de hardware, como protección de memoria; temporizador, para controlar la gestión de los tiempos de trabajo; instrucciones privilegiadas, que permite al monitor declinar o permitir algunos procesos; e interruptores, que enfatiza en el control que puede tener el monitor.

1.3 Sistemas de multiprogramación de lotes.

Para la segunda mitad de la década de los 60's, surgen los circuitos integrados junto con la multiprogramación, que se basa en la división de la memoria para la asignación de distintos trabajos a cada parte. La IBM 360 fue la primera línea de computadoras en utilizar circuitos integrados, lo que hizo dispositivos más completos y eficientes en comparación a los de la segunda generación. Además, los sistemas operativos en esta generación tenían la capacidad de leer más rápidamente trabajos en las tarjetas y ejecutarlos, para que apenas terminase un trabajo en ejecución, se pudiese cargar un nuevo trabajo al disco, se le llamó spooling a este método.

El spooling tiene la habilidad de mantener en una estructura de datos a todos los Jobs listos para ser ejecutados. La estructura permite seleccionar a cualquier Job, posibilitando un mejor control y manejo del uso de la Unidad Central de Procesamiento. Cuando es seleccionado un Job para su ejecución es llamado schedulling, que es lo que permite la multiprogramación. La multiprogramación permite una organización en la Unidad Central, permitiendo que siempre pueda realizar algún proceso.

Sabemos que debe de haber suficiente memoria para contener el Sistema Operativo (monitor residente) y un programa de usuario. Si hay espacio para el SO y dos programas de usuario, mientras que un trabajo necesita esperar a la entrada o salida, el procesador puede cambiar al otro, que puede no esperar entradas o salidas. A su vez, podríamos extender la memoria para contener tres, cuatro o más programas y que cambien entre ellos. Este enfoque es conocido como multiprogramación, o multitarea. Es el tema central de los sistemas operativos modernos.

También surgieron conceptos importantes con respecto al tiempo simultáneo en el que se podían realizar trabajos en las máquinas, donde distintos usuarios podían tener acceso a terminales en línea. Fue el Compatible Time Sharing System (CTSS) el primer sistema en cumplir con las características anteriormente mencionadas.

1.4 Tiempo Contemporáneo.

Para la década de 1980, Internet tuvo su crecimiento. A su vez, los sistemas operativos sufrieron una gran evolución. Para la siguiente década surgen conceptos como la computación distribuida y el multiprocesamiento a través de distintas redes computacionales. Fue una época donde se enfocaron mucho en la conectividad, siendo facilitada gracias a protocolos y estándares que surgieron. En esta época se le empezó a tomar importancia a la experiencia de un usuario al utilizar algún programa, dando lugar a software amigables para el usuario.

El crecimiento de las redes dio lugar a sistemas operativos de redes, que conectaban a varias computadoras. Individualmente, los dispositivos poseían su propio sistema operativo. Estos sistemas operativos pueden controlar interfaces de redes y software de bajo nivel, además de proveer software para conexión y acceso a información, archivos, etc.

Otra aparición fue los sistemas operativos distribuidos, facilitando la vida de los usuarios ya que se podían olvidar por completo de ubicación de archivos o ejecución de programas, siendo el SO el encargado de gestionar automáticamente todos estos procesos. Gracias a los circuitos integrados, ahora podía haber procesadores con miles de transistores en un centímetro cuadrado de silicio. Esto provoca la posibilidad de construir computadoras personales.

El desarrollo de Internet y los sistemas operativos ha sido un proceso dinámico y complejo, que ha implicado cambios tecnológicos, sociales y culturales. Desde los primeros sistemas operativos por lotes hasta los actuales sistemas operativos distribuidos, se ha buscado mejorar el rendimiento, la eficiencia, la seguridad y la experiencia de uso de las computadoras. La conectividad y la cooperación entre distintos dispositivos y redes ha sido un factor clave para el avance de la computación. La experiencia del usuario ha sido relevante para diseñar y evaluar los sistemas operativos. Así, los sistemas operativos han evolucionado junto con las necesidades y demandas de los usuarios, adaptándose a los nuevos desafíos y oportunidades que ofrece la tecnología.

2. Sistemas Operativos y sus objetivos.

Un programa es una serie de instrucciones almacenadas en algún sitio. En el caso de las computadoras, son almacenadas en la memoria. El procesamiento de una instrucción iniciando cuando el procesador la lee de la memoria, proceso llamado fetch, y ejecuta cada instrucción. La ejecución de un programa se basa en la repetición del fetch de instrucciones y ejecución de instrucciones, provocando esto último una serie de subprocesos necesarios. Todo el proceso necesario para ejecutar una instrucción es llamado ciclo.

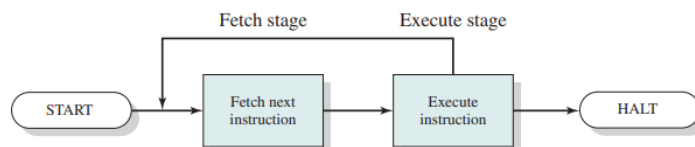


Figura 3. Ciclo Básico de una Instrucción.

La instrucción que fue cargada de la memoria es cargada en un registro de instrucciones, cada instrucción contiene en su interior bits que indican específicamente qué función debe cumplir. El procesador interpreta dichos bits y desempeña la instrucción en cuestión. No obstante, durante este proceso puede haber interrupciones o errores.

Ahora bien, todo el ciclo consta de muchos procesos en los que se administra la memoria, el procesador y las instrucciones, entre muchas otras cosas. Si el usuario tiene que preocuparse constantemente de todos estos aspectos, la eficiencia que puede darle el usar una computadora se ve reducida ya que tiene que dedicarle mucho tiempo al ciclo. Es por ello por lo que el intentar hallar una herramienta que sea capaz de gestionar todo ello fue de suma importancia para los desarrolladores.

Es complicado definir el término de Sistema Operativo, ya que realmente abarca muchas cosas. Un Sistema Operativo, o SO, es un programa que funge como un intermediario entre las aplicaciones y programas que puede tener un sistema, y su hardware, permitiendo que el usuario trabaje en un entorno más amigable y comprensible. El SO se encarga de administrar los recursos que pueda tener el sistema. De manera primitiva, la arquitectura de las máquinas y los recursos que pueden proveer son complejos de manejar, un SO hace que el individuo se pueda despreocupar de ello, ayudando a que la computadora sea una herramienta.

Es una realidad que un SO es un programa, pero uno sumamente complejo e importante. Al encender una computadora, el Sistema Operativo se carga en la memoria y se ejecuta después. Un SO debe ser conveniente, ya que su función es facilitar el uso de una computadora, no perjudicarlo; además debe ser eficiente, ya que debe proveer la capacidad de administrar los recursos correctamente; y debe tener la habilidad de envolver al usuario, ya que el usuario debe sentirse seguro de que puede desenvolver sus trabajos en la máquina.

Un SO tiene la capacidad de crear, eliminar y utilizar objetos del software, como procesos y archivos. Un sistema operativo es el que despierta a la máquina, haciendo la conexión entre procesador, memoria, unidades de disco y cualquier otro dispositivo conectado, esto es importante para verificar que no hay errores, este proceso es llamado POST. Por lo anteriormente expuesto es que un Sistema Operativo es un programa fundamental en una computadora que actúa como intermediario entre el usuario y el hardware del sistema, permitiendo una interacción más amigable y facilitando el uso de la computadora.

Un sistema operativo se basa en procesos, que son programas en ejecución. Un proceso abarca el programa ejecutable, datos y pila, contador y diferentes registros, además de datos necesarios para el correcto desempeño del programa. Hay otro apartado, tabla de procesos, donde se almacena la información de control de los programas. Esta tabla es administrada por el sistema operativo. Un proceso siempre ocupa un espacio en memoria y datos obtenidos de la tabla de procesos. Además, un proceso puede tener subprocesos en sí, formando árboles de procesos.

Los SO también pretenden dar independencia a los computadores, surgiendo así el concepto de directorio, cómo el sistema agrupa y clasifica los archivos. La manera de estructuración de directorios es jerárquica, por ende, a cada archivo se le es asignada una ruta de acceso. Otro objetivo que puede poseer un SO es llamar al sistema, proveyendo a los programas la capacidad de comunicarse con él y solicitar servicios.

2.1 Estructura del Sistema Operativo.

Un Sistema Operativo tiene características internas y externas, por lo que se puede clasificar considerando estos criterios. También, se pueden distinguir las distintas partes de un SO como niveles. Se considera el primer nivel como Gestión del procesador. Es aquí donde yace el Kernel o núcleo, sección encargada de la gestión de la Unidad Central de Procesamiento (CPU). Es cargado en memoria a la hora del encendido, y desempeña sus funciones en esta zona.

Las principales tareas del Kernel es el control de las interrupciones y manejo de condiciones de error, dos aspectos cruciales ya que el sistema puede enfrentarse constantemente a errores o fallas. Además, es el encargado de la comunicación y cambio de procesos, llevando la administración de los procesos que están en activo, asignando la cantidad de recursos proveniente del CPU necesaria para cada proceso. Esta comunicación suele realizarse mediante semáforos, que es una herramienta de sincronización, o mensajes.

El segundo nivel es el encargado de la Gestión de memoria. Es bien sabido que la memoria de las máquinas es limitada. Los procesos requieren memoria para su ejecución. Aquí surge la necesidad de tener un gestor de memoria, que es el encargado de asignar la cantidad de memoria para cada proceso. Es capaz de asignar y liberar memoria, también puede controlar las violaciones de acceso a zonas de memoria a la que el usuario no tiene acceso.

El tercer nivel se enfoca en la Gestión de procesos. Los procesos pueden ser creados y terminados. Este gestor es el encargado de esa tarea, intercambiando mensajes y pudiendo identificar cuando está arrancando algún proceso. Las máquinas están interactuando frecuentemente con dispositivos que permiten la entrada y salida de datos. Es por ello por lo que el cuarto nivel es el encargado de gestionar estos dispositivos, teniendo el nombre de Gestión de dispositivos (E/S). Este nivel se encarga de administrar las entradas y salidas del dispositivo, siendo capaz de crear procesos que lean e interpreten estas entradas y salidas, a su vez, pueden controlar los dispositivos E/S directamente, y planificar sus tareas.

El quinto nivel es el encargado de la Gestión de la información. Es el nivel capaz de administrar el acceso de recursos, realizado mediante nombres lógicos. Estos identificadores permiten identificar con mayor facilidad rutas de acceso que pueden ser complicadas de interpretar, siendo el SO el encargado de controlar esta información. Este nivel también pretende proteger la información, creando y destruyendo ficheros y directorios, además de protección de acceso.

Cada nivel posee subfunciones igualmente importantes, además de tener la habilidad de utilizar funciones información brindada por niveles previos. El desarrollo de SO ha permitido crear interfaces que permiten que los usuarios tengan mayor accesibilidad a la hora de usar alguno de estos gestores, sin necesidad de perjudicar al resto.

Otro apartado importante en un SO es el Shell, o intérprete de comandos. El SO es el software que, entre muchas funciones, permite hacer llamadas al sistema. El Shell no es particular de los SO, pero si tienen una estrecha relación, ya que utiliza muchas características del Sistema Operativo. Un Sistema Operativo tiene, además de los apartados mencionados, tiene muchas otras características y aspectos que hacen que éste logre sus objetivos, permitiendo que el usuario tenga una mejor experiencia a la hora de utilizar sus dispositivos.

2.2 Funciones del Sistema Operativo.

Las principales funciones de un SO es controlar la ejecución de los programas, aceptando trabajos y administrando su ejecución; administrar periféricos, coordinando y manipulando

aquellos dispositivos que se conecten a la computadora; gestiona permisos de usuario, evitando que haya acciones que afecten a otros; desarrollo de programas, ya que provee herramientas que ayudan a desarrolladores a crear nuevos programas o mejorar los ya existentes.

Una función crucial es el acceso a archivos, ya que el SO analiza todos los datos de la computadora y restringe acceso si se está en un sitio de trabajo con personal. El SO también provee detección de errores, ya que es una realidad que las computadoras no están desprovistas de errores, pudiendo ser internos o externos, como un error de memoria. El SO debe clasificar y responder si se presenta un error, puede ir desde cerrar el programa hasta reportar el error.

Un buen Sistema Operativo, a su vez, tiene la capacidad de analizar el uso que le da el usuario y recolectar estadísticas de varias partes de la máquina, como sus recursos, para evaluar su desempeño. Esto es vital para la constante mejoría de los dispositivos y concretamente del SO. La arquitectura del dispositivo cuenta con un set de instrucciones, llamado ISA. El ISA es el que determina qué instrucciones puede ejecutar la computadora. El Sistema Operativo es también un intermediario entre este set de instrucciones y las aplicaciones, teniendo acceso al lenguaje de la máquina.

Un sistema operativo pretende restar carga del trabajo al usuario, haciendo que la experiencia de uso del ordenador sea la de mayor calidad posible. A su vez, desempeña múltiples funciones esenciales en una computadora. Desde controlar la ejecución de programas y administrar periféricos, hasta gestionar permisos de usuario y desarrollar programas, siendo fundamental para el funcionamiento eficiente y seguro de una computadora.

2.3 Modos del Sistema.

Debido a la importancia que tiene un Sistema Operativo en el funcionamiento de un dispositivo, surgieron dos modos de operación, el modo usuario y el modo supervisor. Cuando la computadora está ejecutando alguna aplicación, el SO está en modo usuario, no obstante, cuando el programa requiere alguna funcionalidad, permiso o servicio entonces hay una transición a un modo supervisor. Cuando se inicializa el sistema, lo hace en modo supervisor, el sistema operativo empieza a funcionar y ejecuta las aplicaciones en modo usuario. Cuando hay alguna interrupción, el sistema vuelve a modo supervisor, habiendo un constante cambio entre ambos modos.

Podemos notar que cuando el SO tiene el control del dispositivo entonces está en modo supervisor, cambiando posteriormente a modo usuario. Esta oscilación entre los modos prevé que se le dé un uso incorrecto al sistema, evitando que pueda ser desconfigurado o se cometa algún error en contra de las instrucciones privilegiadas. El hardware solo permite que las instrucciones privilegiadas se ejecuten en el modo supervisor, y no responde en el caso de que se intente acceder a alguna de estas instrucciones en modo usuario. La instrucción de cambiar a modo supervisor es un ejemplo de una privilegiada.

2.4 Tipos de Sistemas Operativos.

Podemos clasificar Sistemas Operativos utilizando diferentes criterios, como lo es la estructura, modos de explotación, servicios que ofrece o la manera en la que los ofrece.

2.4.1 Sistemas Operativos por su estructura.

Para este apartado influye mucho el diseño del SO. A la hora de la realización de un sistema se toma en cuenta los requisitos del usuario, ya que éste espera un sistema que sea fácil de utilizar, o al menos fácil de aprender a usarlo, además de seguro, rápido y agradable, permitiendo desempeñar distintas actividades dependiendo de las necesidades del cliente. Pero no es lo único importante, ya que también el software tiene ciertos requisitos, como lo es el mantenimiento, cómo opera, cuáles son las restricciones de uso, eficiencia, manejo de interrupciones o errores y flexibilidad.

Existen distintos tipos de estructura, iniciando por la monolítica. Era muy usada en los inicios de los SO. Estaba conformada de un solo programa, el cual poseía varias rutinas que tenían la capacidad de comunicarse entre sí. Este tipo de estructura se caracterizaba por una buena definición de parámetros de enlace entre las mencionadas rutinas. Carecía de protecciones y privilegios cuando llegaba el momento de entrar a rutinas que manejan los recursos de la máquina, siendo este uno de sus puntos débiles. Era una estructura que tenía únicamente modo supervisor.

Posteriormente surgió la estructura jerárquica, dividiendo al sistema operativo en pequeñas partes, definiendo claramente cada una. Es aquí donde se encuentran los niveles de gestión vistos previamente. Una máquina virtual es también una estructura. Éstas muestran una interfaz a cada proceso, separando dos conceptos que suelen estar unidos, la multiprogramación y la máquina extendida. Los usuarios buscan juntar varios sistemas operativos utilizando las máquinas virtuales, llamando al núcleo monitor virtual, que tiene la función de llevar a cabo la multiprogramación.

Existe otro tipo de estructura llamada cliente-servidor. Es útil para cualquier tipo de programa, además de cumplir con procesos que suelen hacer SO convencionales. Gestiona adecuadamente los procesos, la comunicación entre ellos y la memoria. Su núcleo está diseñado para una correcta comunicación entre clientes y servidores.

2.4.2 Sistemas Operativos por modos de explotación.

Se trata de las funcionalidades que se le puede dar a un sistema operativo. En este tipo entra el procesamiento por lotes, explicado previamente. Ejemplos de este tipo de sistemas son el SCOPE y el EXEC II. Otra modalidad de explotación que puede tener un SO es la multiprogramación, también explicada con anterioridad. En este apartado, el SO se encarga de

distribuir la carga computacional entre los procesos que se estén desempeñando en el momento, permitiendo que la máquina tenga mayor poder de procesamiento.

2.4.3 Sistemas Operativos por los servicios ofrecidos.

Es la clasificación de SO que tiene más en cuenta al usuario. Puede dividirse por el número de usuarios, número de tareas o número de procesadores. Iniciando con el número, existen los SO monousuario, que como su nombre expresan, están diseñados para ser utilizados por un único usuario a la vez. Existe también los SO multiusuarios, que dan servicio a 2 o más personas a la vez, pudiendo ser mediante terminales conectadas a la computadora o gracias a sesiones en una red.

Dependiendo del número de tareas puede ser un SO monotareas, que permiten la ejecución de una tarea a la vez. También están los SO multitareas, que tienen la capacidad de desempeñar más de una tarea a la vez, se suelen caracterizar por el uso de menús y de un mouse, permitiendo navegar con facilidad entre las tareas.

Por último, dependiendo de la cantidad de procesos, están los SO uniproceto, que permiten realizar un proceso a la vez, pudiendo simular multitarea haciendo que se realice una tarea y, al ejecutar otra, se intercambié el foco del SO. También están los multiprocesos, que permiten realizar mayor cantidad de procesos al mismo tiempo. En esta clasificación existe la subdivisión de simétricos, que distribuyen la carga de procesamiento por igual, y los asimétricos, que asignan una tarea a cada procesador existente, en esta clasificación es la que entra Windows.

2.4.4 Por la manera de ofrecer los servicios.

En este apartado se puede dividir en sistemas centralizados, en los cuales los computadores se encargaban de todo el procesamiento, mientras que los usuarios se hacían cargo únicamente de las terminales brutas. También están los sistemas de red, que son aquellos que conectan dispositivos mediante algún medio de comunicación, buscando facilidad a la hora de compartir los recursos y la información del sistema.

Otro tipo de sistema son los sistemas distribuidos. Permiten distribuir trabajos, tareas y procesos entre una serie de procesadores, pudiendo estar en la misma máquina o en alguna diferente. Tiene una subdivisión, donde entre un sistema fuertemente acoplado, que comparte la memoria y un reloj global para registrar tiempos de acceso, y un sistema débilmente acoplado, en el que los procesadores no comparten la memoria ni otros aspectos. También están los sistemas operativos de tipo paralelo, de gran utilidad en casos en los que existan varios procesos que demanden el mismo recurso, y que puedan ser ejecutados al mismo tiempo.

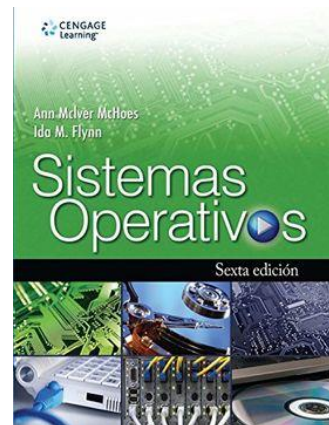
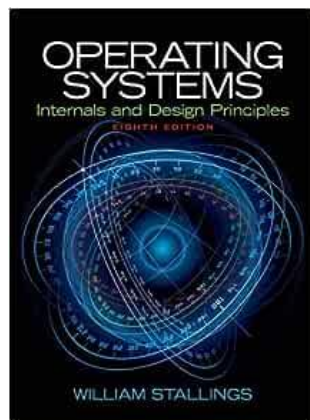
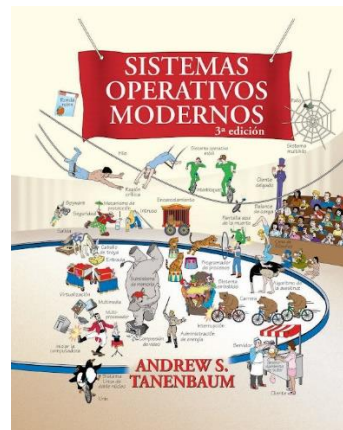
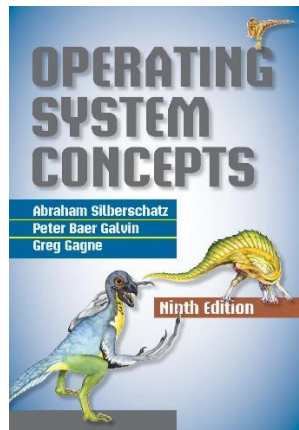
Los sistemas operativos son programas fundamentales para el funcionamiento de las computadoras, ya que gestionan los recursos de hardware y software, y proveen una interfaz

para el usuario. Como se pudo observar, determinar si hay un sistema operativo mejor que otro es algo completamente subjetivo. La segmentación y clasificación de los SO se ha hecho precisamente para que cada usuario pueda identificar a dónde quiere llegar con la tarea que pretende desempeñar, y consecuentemente elegir el sistema operativo que se adapte a sus necesidades. Cada tipo de sistema operativo tiene sus ventajas y desventajas, y se adapta mejor a ciertas aplicaciones o necesidades.

3. Preguntas.

3.1 Temas en común en los libros buscados.

Primero que nada, es importante identificar los libros utilizados, que son los siguientes:



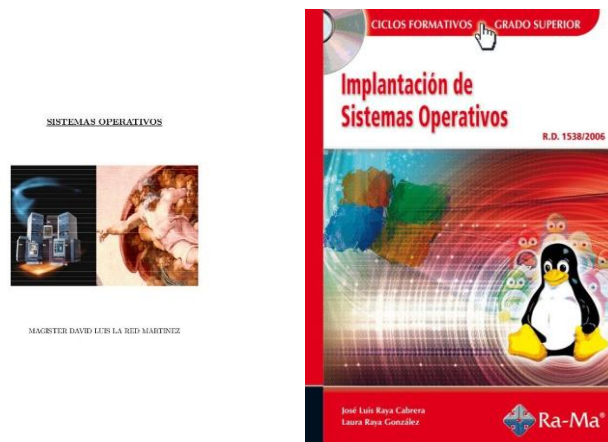


Figura 4. Bibliografía empleada para la investigación.

Todos tenían una estructura similar, empezando por la definición de los sistemas operativos. Algunos se basaban más en citar definiciones que han dado personajes a lo largo de la historia, otros eran más conceptuales e iban más al grano. El único que tuvo un inicio diferente fue Operating System de William Stalling, que inicia explicando a grandes rasgos como es el funcionamiento de una computadora, desde que es encendida hasta la ejecución de tareas. Algunos de los temas que tienen en común son:

- Definición de los sistemas operativos.
- Estructura de los sistemas operativos.
- Pequeño acercamiento al funcionamiento de una computadora.
- Funciones de un sistema operativo.
- Capacidades de un sistema operativo.
- Característica de los sistemas operativos.
- Pequeño acercamiento a la arquitectura de computadoras.

Cada libro tenía un enfoque diferente, pero estos fueron los temas que se tocaron en todos. Algunos otros temas que se tocaron en la mayoría fueron:

- Evolución de los sistemas operativos.
- Procesamiento secuencial.
- Procesamiento por lote.
- Multiprogramación.
- Llamadas al sistema.
- Shell.
- Ciclo de una instrucción (fetch y ejecución).
- Tipos de sistemas operativos.

Algunos libros eran mucho más técnicos, otro más conceptuales. Haciendo un añadido, los 2 que más me gustaron fueron Operating System de William Stallings y el de Implementación de Sistemas Operativos de José Luis Reyes y Laura Raya.

3.2 ¿Qué es un archivo por lotes?

El sistema de procesamiento por lote fue un avance importante en los sistemas operativos. Se basaba en la acumulación de trabajos de la misma índole para posteriormente realizarlos, ahorrando tiempo cuando las computadoras tardaban mucho en realizar sus tareas. Llevándolo a los archivos, un archivo por lote es uno que contiene una serie de instrucciones que serán ejecutadas tal como se encuentran escritas, permitiendo que las tareas se realicen sin necesidad de ser supervisadas, tal como se logró cuando surgió el sistema de procesamiento por lotes. Este tipo de archivos tienen sus instrucciones específicas, y tienen una extensión “.bat”.

3.3 Explicación de procesamiento en serie.

El sistema de procesamiento en serie se basa en que cada proceso debe esperar al proceso anterior para ejecutarse. Es decir, los procesos se van ejecutando uno por uno hasta completar toda la tarea. De allí es donde proviene el nombre, ya que utiliza una organización secuencial, ya que no existe la posibilidad de hacer ejecutar de un proceso simultáneamente. Esto viene de las computadoras antiguas, en la que un usuario debía esperar a que otro complete sus labores para poder usar la computadora, ya que solo había una y se tomaba mucho tiempo en completar los procesos. Como se puede intuir, este sistema de procesamiento es tedioso, ya que cada proceso puede tomarse su tiempo, y como es secuencial, no hay manera de ir adelantando en otros procesos mientras uno se está ejecutando.

3.4 Definición de Sistema Operativo.

Un Sistema Operativo es el medio que tiene el usuario para establecer una comunicación con el hardware del dispositivo que esté utilizando, teniendo la seguridad de que cualquier cosa que realice no afectará al funcionamiento de la máquina. Esto debido a que el SO es el encargo de poner las limitantes de acceso que tendrá el usuario. Un SO es un software complejo que tiene la capacidad de gestionar muchos aspectos que anteriormente debía hacer el usuario, como la administración de memoria y procesos, archivos, permisos, etc. Lo que provee el SO es una positiva experiencia de uso del dispositivo, siendo agradable y ameno para la persona que lo use.

3.5 Lista de los gestores del Sistema Operativo.

- Gestor de procesamiento: Trabaja a nivel de memoria, y es crucial en interrupciones de memoria o cualquier inconveniente que pueda suscitar a la hora de ejecutarse algún programa. Tiene el deber de distribuir recursos provenientes de la Unidad Central de Procesamiento.

- Gestor de memoria: Es el gestor encargado de asignar de manera adecuada la cantidad de memoria que requiere cada proceso en ejecución, esto debido a que la memoria es limitada y no puede despilfarrarse.
- Gestor de E/S: Un dispositivo que no tenga otros elementos de entrada y salida de datos es un dispositivo muy limitado. La entrada y salida de datos es una de las principales razones por las que se usan tanto las computadoras. Este gestor es el encargado de que haya una buena comunicación entre dispositivos de entrada y salida con la computadora.
- Gestor de procesos: Este, a diferencia del de procesamiento, no es el encargado de distribuir los recursos del CPU, pero si se encarga de determinar el orden con el que deben ser ejecutados los procesos, usando su capacidad de interrumpir o iniciar cualquier programa.
- Gestor de información: Se encargar en ofrecer una experiencia intuitiva por el usuario a la hora de acceder a archivos y rutas, entendiendo que debe almacenar los archivos para cualquier momento que el usuario lo necesite.
- Todos los gestores mencionados tienen la capacidad y el deber de comunicarse entre sí, yendo desde el gestor más cercano al hardware hasta el más alejado, para que haya un correcto funcionamiento de la máquina, y haciendo énfasis en la seguridad y protección, ya que el SO debe encargarse de evitar que una persona sin permiso pueda acceder a archivos importantes para el usuario.

3.6 Objetivos de un Sistema Operativo.

Un Sistema Operativo tiene como objetivo evitar que el usuario se preocupe por cuestiones que atañen al hardware y al lenguaje máquina, gestionando de manera eficiente los recursos tanto de procesador como de memoria, evitando que el usuario acceda a zonas prohibidas que puedan perjudicar el uso del computador. Además, el SO busca que el usuario se sienta cómodo a la hora de usar una computadora, para así sentir que es una herramienta más que un problema, proporcionando una interfaz intuitiva y sencilla, que cualquier persona de cualquier contexto pueda utilizar.

Por la misma inexperiencia que pueden tener algunos usuarios, otro de los objetivos de un SO es limitar las zonas de la máquina a las que se tienen acceso, creando modos como el de usuario y de supervisor, que permiten ejecutar tareas dependiendo de la importancia que tengan éstas y cuánta comunicación tengan con el hardware. Además, un SO debe permitir que, en caso de haber una falla, el usuario pueda solucionarlo y seguir usando su computadora con normalidad. Entendiendo así que el objetivo principal de un SO es que el usuario pueda sacar el mayor provecho de una máquina, sin estropearla durante su uso, y disfrutando al máximo de sus capacidades.

3.7 Ilustración y explicación del ciclo Fetch.

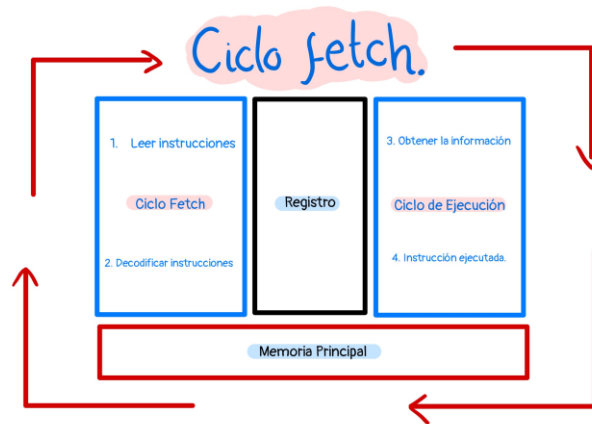


Figura 5. Ilustración Ciclo Fetch junto con Ciclo de Ejecución.

El ciclo Fetch trabaja en conjunto con el ciclo de ejecución, siendo estos ciclos los necesarios para la realización de procesos. El ciclo fetch se basa en la lectura y decodificación de la instrucción que será ejecutada. Posterior a la lectura, la instrucción es registrada en el Registro de Instrucciones, para posteriormente ser ejecutada, realizando las tareas que indique la instrucción. Este ciclo es repetitivo, ya que ejecuta las instrucciones que haya en el registro, deteniéndose cuando no haya más instrucciones o haya alguna señal de detención.

3.8 Clasificación de los sistemas operativos.

Los sistemas operativos han evolucionado mucho y han sido capaces de adaptarse a los diferentes usos que puede darle un usuario, es por ello por lo que hay distintos criterios de clasificación. Una clasificación es por su estructura, que puede ser monolítica, que fueron los primeros tipos de SO, que se basaban en una constante comunicación de rutinas; jerárquica, que son los SO que delimitan muy bien sus partes y le asignan tareas específicas a cada una; máquina virtual, que son SO que buscan la conexión entre muchos SO; y cliente servidor, que se basa en una correcta gestión de memoria, procesos y la comunicación entre estos.

También pueden clasificarse por el modo de explotación, donde entra el procesamiento por lote, que se basa en la acumulación de trabajos similares para realizarlos lo más pronto posible; y la multiprogramación, que se basa en la simultaneidad de procesos. Los servicios ofrecidos también es un criterio para seguir, pudiendo ser por el número de usuarios, uno o más de uno; el número de tareas, una o más de una; y el número de procesadores, ya sea un proceso o multiprocesos. Todos los SO que conocemos pueden clasificarse en una o más de las clasificaciones mencionadas.

4. Archivo Por lote.

En este apartado se muestra el archivo por lote con 10 instrucciones estudiado para la investigación:

```
1  @echo off
2  echo archivo por lote para la Actividad 1. Diego Maldonado. Sección D04
3
4  echo 1. Instruccion: Listando archivos en el directorio
5  dir
6
7  echo 2. Instruccion: Crear nueva carpeta
8  mkdir NuevaCarpeta
9
10 echo 3. Instruccion: Cambiar al directorio de la nueva carpeta
11 cd NuevaCarpeta
12
13 echo 4. Instruccion: Crear un archivo de texto
14 echo Diego Maldonado Sistemas Operativos Prueba | > archivo.txt
15
16 echo 5. Instruccion: Mostrar el contenido del archivo
17 type archivo.txt
18
19 echo 6. Instruccion: Regresar al directorio anterior
20 cd..
21
22 echo 7. Instruccion: Eliminar la carpeta creada
23 rmdir /s /q NuevaCarpeta
24
25 echo 8. Instruccion: Mostrar información del sistema
26 systeminfo
27
28 echo 9. Instruccion: Mostrar la fecha y hora actuales
29 echo %date% %time%
30
31 echo 10. Instruccion: Finalizar la ejecución
32 echo Prueba finalizada.
33 pause
```

Figura 6. Codificación del archivo por lote con 10 instrucciones.

La instrucción echo off impide que se muestren comandos a la hora de la ejecución. El comando echo es una impresión en pantalla de la cadena de caracteres que tenga a continuación. La 1º instrucción, que es dir, se basa en mostrar los archivos que haya en el directorio en el que se encuentre el archivo .bat; la 2º, que es mkdir permite crear una nueva carpeta; la 3º, cd, permite ir al directorio de la carpeta cuyo identificador se escriba a continuación. La 4º instrucción crea un archivo .txt y le escribe el texto que posee. La 5º muestra qué contiene ese archivo .txt. La 6º, que es cd., que provoca un retroceso en el directorio. La 7º, que es rmdir con las opciones /s, que indica que se eliminará el directorio y los archivos de manera recursiva, y /d, que indica que no se requerirán confirmaciones del usuario. La 8º fue la que más me sorprendió, ya que muestra toda la información de mi sistema, eran líneas y líneas de

información. La 9° es un echo pero con los comandos %date% y %time% que muestra fecha y horas, y para finalizar con un echo que confirma que todo se ejecutó correctamente.

```

1. Instruccion: Listando archivos en el directorio
El volumen de la unidad C es Windows
El número de serie del volumen es: AC74-5587

Directorio de C:\Users\diego\OneDrive\Escritorio\Tareas\5to semestre\Sistemas Operativos\archivo por lote

26/08/2023  06:51 p. m.    <DIR>        .
26/08/2023  06:33 p. m.    <DIR>        ..
26/08/2023  06:52 p. m.                851 lote.bat
                1 archivos                851 bytes
                2 dirs 37,870,100,480 bytes libres
2. Instruccion: Crear nueva carpeta
3. Instruccion: Cambiar al directorio de la nueva carpeta
4. Instruccion: Crear un archivo de texto
5. Instruccion: Mostrar el contenido del archivo
Diego Maldonado Sistemas Operativos Prueba
6. Instruccion: Regresar al directorio anterior
7. Instruccion: Eliminar la carpeta creada
8. Instruccion: Mostrar informaci|n del sistema

Nombre de host:                ENIAC
Nombre del sistema operativo:   Microsoft Windows 11 Home Single Language
Versión del sistema operativo:  10.0.22000 N/D Compilación 22000
Fabricante del sistema operativo: Microsoft Corporation
Configuración del sistema operativo: Estación de trabajo independiente
Tipo de compilación del sistema operativo: Multiprocessor Free

```

Figura 7. Muestra del funcionamiento del archivo por lotes.

El archivo se ejecutó de manera satisfactoria, permitiéndome entender que son una serie de instrucciones y comandos que algunos ya conocía de haber usado el Shell de Windows, sin embargo, no sabía que había manera de automatizarlos en forma de archivos por lote.

5. Conclusiones.

El concepto de Sistema Operativo no ha existido desde el inicio de las computadoras, pero desde siempre se ha sabido que es necesario. Su concepción fue concretamente en la década de 1950, por la necesidad que tenían los usuarios de aquella época de controlar el funcionamiento y el uso de recursos de las computadoras. Desde entonces, los SO únicamente han evolucionado y mejorado. Su principal objetivo es simplificar la manera en el que el usuario utiliza su máquina, buscando en todo momento ayudar a controlar los aspectos que, en años anteriores, eran muy complejos para el usuario, como saber desde sistema binario y asignación de espacios en memorias. Un Sistema Operativo no es más que un mediador entre nosotros y el hardware, facilitando la interacción y control de los componentes de la computadora. El uso de un sistema operativo no solo permite un control más fácil y eficiente de las tareas en un dispositivo, sino que también mejora la seguridad, estabilidad y rendimiento del sistema en general.

Es muy interesante el cómo se llegó hasta los avances que tenemos en la actualidad, ya que los desarrolladores hicieron una correcta abstracción de como realizamos los procesos en el mundo, y como pueden ser realizados por las computadoras, generando un entorno más amigable y accesible para los usuarios. Tan profundos han sido los avances que ahora existen distintos sistemas operativos dependiendo del tipo de usuario que seamos, así como de la tarea que busquemos desempeñar, pero nunca dejando de lado el principal objetivo, que es la comodidad de uso, así como prevenir y limitar las acciones de usuario, para así proteger a los dispositivos de usos incorrectos. Los sistemas operativos son el resultado de un largo proceso de innovación y adaptación a las necesidades de los usuarios de las computadoras. Evolución que debemos agradecer, ya que, gracias a ella, hemos logrado simplificar y mejorar nuestra interacción con las computadoras.

6. Referencias.

Flynn, I. M., & McHoes, A. M. (2001). *Sistemas operativos*. Cengage Learning Latin America.

La Red Martínez, D. L. (2004). *Sistemas operativos*. El Cid Editor. <https://elibro-net.wdg.biblio.udg.mx:8443/es/lc/udg/titulos/77467>

Raya Cabrera, J. L. (2015). *Implantación de sistemas operativos*. RA-MA Editorial. <https://elibro-net.wdg.biblio.udg.mx:8443/es/lc/udg/titulos/62453>

Silberschatz, A., Galvin, P. B., & Gagne, G. (2012). *Operating system concepts*. Wiley.

Stallings, W. (2015). *Operating systems: Internals and Design Principles*. Pearson Prentice Hall.

Tanenbaum, A. S. (2008). *Sistemas Operativos Modernos*. Pearson Prentice Hall.