Fundamentos de sistemas operativos

# Evolución Histórica de los Sistemas Operativos:

* 1.ª Generación (1945-1955): Sin sistemas operativos. Programación directa en código máquina mediante paneles de cableado y luego tarjetas perforadas.
* 2.ª Generación (1955-1965): Introducción de transistores, procesamiento en lotes, y control mediante tarjetas perforadas.
* 3.ª Generación (1965-1980): Surgimiento de circuitos integrados, sistemas de tiempo compartido, multiprogramación y nuevos lenguajes de programación.
* 4.ª Generación (1980-actualidad): Generalización de circuitos integrados, llegada de computadoras personales, sistemas operativos de tiempo real y multitarea.

# Componentes de los Sistemas Operativos:

## Elementos:

* Núcleo (Kernel): Coordina el sistema, gestionando interrupciones, asignación de tareas y comunicación entre programas. Destaca el papel del planificador.
* Administrador de Memoria: Gestiona la memoria principal (RAM) y aplica técnicas como la memoria virtual para un mejor aprovechamiento.
* Gestión de Entrada y Salida: Asegura la presentación o almacenamiento de datos, adaptando las peticiones al dispositivo específico utilizado.
* Administrador de Archivos: Responsable de la estructura de datos y programas de los usuarios, supervisando la creación, modificación y eliminación de archivos y carpetas.

## Estructura:

* Monolítica: Primeros sistemas, un solo programa sin estructura clara.
* Jerárquica o por Capas: Fragmentación del sistema en partes más pequeñas y especializadas para facilitar el mantenimiento.
* Microkernel o Cliente/Servidor: Organización en módulos aislados del núcleo, ofreciendo mayor tolerancia a fallos, seguridad y portabilidad.

# Clasificación y Tipos de Sistemas Operativos

* En esta sección, se aborda la clasificación de los sistemas operativos según diversas características. La elección del tipo de sistema operativo depende del uso que se le dará. A continuación, se presentan los criterios de clasificación:

## Según la utilización de los recursos:

* Mono programados: Solo admiten un programa cargado en el sistema, ejecutándose hasta su finalización sin realizar otras tareas.
* Multiprogramados: Capaces de realizar varias tareas al mismo tiempo, con varios programas cargados en memoria.

Importante: Se distingue entre programa y proceso. Un programa es un conjunto de instrucciones en el disco duro, mientras que un proceso es un programa en ejecución en la memoria.

* Multitarea apropiativa: El sistema decide qué proceso utiliza el procesador en cada momento.
* Multitarea cooperativa: El propio proceso decide si continúa utilizando el procesador o lo cede a otro.

## Según su interactividad:

* Procesamiento por lotes (batch): Se introducen acciones o pasos que el sistema procesa en el orden recibido, generando la salida correspondiente.
* Tiempo compartido (time sharing): Permite la ejecución de varias tareas simultáneamente asignando el procesador a cada tarea por un tiempo limitado.
* Sistemas en tiempo real (real time): Destinados a tareas críticas con límites de tiempo de respuesta conocidos, como el control de vuelo de aeronaves.
* Sistemas en tiempo diferido: No críticos en tiempo de respuesta, más comunes.

## Según el número de usuarios:

* Monousuario: Un solo usuario puede usar el sistema a la vez.
* Multiusuario: Varios usuarios pueden acceder al sistema simultáneamente mediante técnicas de multiprogramación.

## Según el tipo de aplicación:

* Propósito general: Capaces de ejecutar cualquier tipo de aplicación informática, ejemplos incluyen Windows, GNU/Linux, Mac OS, Android, iOS, etc.
* Propósito específico: Diseñados para ejecutar aplicaciones específicas, como sistemas operativos en máquinas industriales o electrodomésticos.

## Según el número de procesadores:

* Monoprocesadores: Un solo procesador ejecuta todas las tareas.
* Multiprocesadores: Varios procesadores permiten la ejecución simultánea de varias tareas.

## Según la distribución de tareas del sistema:

* Sistemas distribuidos: Múltiples máquinas se comunican entre sí, compartiendo tareas para mejorar el rendimiento.
* Sistemas centralizados: Una sola máquina realiza todas las tareas del sistema operativo.

# Funciones de un Sistema Operativo

## Funciones del Sistema Operativo:

* El sistema operativo es esencial para el funcionamiento de cualquier sistema informático. Establece las bases para la ejecución de aplicaciones y programas de los usuarios. Aquí se detallan los objetivos y funciones clave de un sistema operativo.

## Objetivos del Sistema Operativo:

* Los objetivos principales son aumentar la productividad de los usuarios, facilitar el uso del equipo y proporcionar un entorno cómodo y abstracto en relación con el hardware. También busca optimizar la utilización del hardware para maximizar el rendimiento y gestionar eficientemente los recursos del hardware y del software.

## Funciones Principales del Sistema Operativo:

* Administrar el procesador o procesadores (en sistemas multiprocesador).
* Administrar la memoria.
* Facilitar la comunicación entre dispositivos y usuarios de manera transparente.
* Organizar datos para garantizar un acceso rápido y seguro.
* Gestionar las comunicaciones de red.
* Facilitar las entradas y salidas.
* Ofrecer técnicas de recuperación de errores.
* Prevenir interferencias entre usuarios.
* Generar estadísticas y archivos de registro del sistema.
* Compartir hardware y datos entre múltiples usuarios.

### Funciones Clave:

* Gestión de Procesos:
  + Estados del proceso: Nuevo, Preparado, En Ejecución, Acabado, Suspensión.
  + Planificación de procesos mediante algoritmos como FCFS, Round Robin, SJF.
  + Uso de Bloques de Control de Procesos (PCB) para gestionar información sobre procesos.
* Gestión de Memoria:
  + Asignación y liberación de memoria para procesos.
  + Control del acceso a la memoria entre procesos.
  + Swapping entre memoria principal y secundaria en casos de ocupación total.
* Gestión de Entrada/Salida (E/S):
  + Interfaz entre periféricos y usuarios.
  + Controladores de dispositivos (device drivers) como intermediarios.
  + Uso de spools y buffers para retener información.
* Gestión del Sistema de Archivos:
  + Diferenciación entre gestión física y lógica de archivos y carpetas.
  + Atributos de archivos como nombre, extensión, permisos, creador, fechas y tamaño.
  + Carpetas como estructuras jerárquicas que organizan archivos.