

SISTEMAS OPERATIVOS Y SOFTWARE DE UN SISTEMA INFORMÁTICO

ACTIVIDAD 1: *Sistemas Operativos: Requisitos hardware, año de aparición, licencia y campos de aplicación.*

Completa la siguiente tabla añadiendo una fila por sistema operativo, utilizándola información más reciente disponible. Asegúrate de incluir sistemas operativos utilizados tanto en entornos de escritorio como en servidores y dispositivos móviles. Además, añade una columna para características de seguridad específicas de cada sistema operativo:

SISTEMA OPERATIVO	PROCESADOR	RAM	ESPACIO DE ALMACENAMIENTO	AÑO DE APARICIÓN	LICENCIA	DISPOSITIVOS DE INSTALACIÓN	CARACTERÍSTICAS DE SEGURIDAD
Windows 11	- Intel Core de 8 ^a gen. y superior. -AMD Ryzen 2000 y superior.	4 GB como mínimo pero mejor 8 GB para un rendimiento óptimo.	64 GB, entre 20 y 30 GB más espacio extra para actualizaciones.	2021	-Microsoft. -EULA (personal). -OEM (vinculada al dispositivo).	-PC (escritorio). -Portátiles. -Tablets.	-Windows defender. -BitLocker. -Control de aplicaciones y navegador.
Windows Server 2022	-Intel Xeon 3 ^o gen y sup. y Core 8 ^o gen y post. -AMD Ryzen 3 ^o gen y sup.	2 GB mínimo pero mejor 4 GB para grandes cargas.	32 GB mínimo para instalar pero lo óptimo es 64 GB.	2021	-Microsoft. -OEM (vinculada al dispositivo) -Licencia estándar.	-Cloud Servers. -Servidores físicos. -Máquinas Virtuales.	-Servidores con núcleos protegidos Cifrado de unidad BitLocker. -Protección contra Ransomware.
Ubuntu Desktop 22.04	-Intel64. -AMD64. -ARM (móviles).	2 GB mínimo pero mejor 4 GB para un mejor uso.	Mínimo 20 GB para instalarse correctamente.	2022	-Licencia Pública General (GPL).	-PC (escritorio). -Portátiles. -USBs.	-Cortafuegos integrado. -Cifrado de disco. -Actualizaciones de seguridad automáticas.
Red Hat Enterprise Linux 9	-CPUs x86_64. -ARM64. -IBM Z.	DDR4 (la más común) LPDDR4/5 (en dispositivos móviles)	Mínimo 6 GB para el sistema operativo base.	2021	-Licencia de uso comercial.	-Cloud Servers (públicas y privadas). -Servidores físicos. -Máquinas virtuales.	-Cifrado de datos (uso de LUKS). -Actualizaciones de seguridad automáticas. -Controles de acceso.
macOS Ventura	-Intel Core 8 ^o gen y post. -Apple Silicon.	4 GB mínimo pero mejor 8 GB para uso.	Mínimo 12 GB pero lo óptimo son 20 GB para actualizaciones.	2022	-Apple. -EULA.	-Mac. -MacBook. -iMac.	-Actualizaciones de seguridad automáticas. -SandBoxing. -Protección del S.O.
Android 13	-Rockchip RK3528, -Quad-Core. -4x ARM Cortex-A53.	2 GB mínimo pero mejor 4 GB para uso óptimo.	Mínimo 5 GB pero lo óptimo son 10 GB para actualizaciones.	2022	-Freeware.	-Teléfonos móviles. -Tablets. -TVs inteligentes.	-Actualizaciones de seguridad. -Cifrado de datos. -Protección contra malware.
iOS 17	-A12 bionic y post. (dispositivos Apple).	2GB mínimo pero mejor 4 GB para uso óptimo.	Mínimo 5 GB pero lo óptimo son 10 GB para actualizaciones.	2023	-Apple. -EULA.	-iPhone. -iPad.	-Seguridad del navegador. -Detección de phishing. -Control de permisos mejorado.
FreeBSD 13	-CPUs x86_64. -ARM.	2 GB mínimo pero mejor 4 GB para uso óptimo.	Mínimo 2 GB para instalación, pero lo óptimo son 10 GB.	2021	-Código abierto.	-Servidores físicos. -Máquinas virtuales.	-Cifrado de disco. -Seguridad de red. -Actualizaciones de seguridad.
Google Fuchsia	-CPUs x86_64. -ARM.	1 GB mínimo pero mejor 2 GB para su optimización.	Mínimo 8 GB para instalación, pero lo óptimo son 16 GB.	2021	-Código abierto.	-PC (escritorio). -Portátiles. -Teléfonos Móviles.	-Cifrado de datos. -Seguridad en la red. -Actualizaciones de seguridad.

ACTIVIDAD 2: Arquitectura interna de un Sistema Operativo.

Para poder conocer el uso de los distintos tipos de núcleos realiza una pequeña comparativa, sobre los sistemas operativos con arquitectura monolítica y arquitectura híbrida y microkernel, rellenando la siguiente tabla:

ARQUITECTURA	VENTAJAS	DESVENTAJAS	S.O. QUE USAN
MONOLÍTICA	<ul style="list-style-type: none"> 1. Facilidad de despliegue y desarrollo. 2. Tiene un desarrollo más rápido y por lo tanto menor complejidad. 3. Menor latencia (no requiere tanta comunicación con otros servicios). 	<ul style="list-style-type: none"> 1. Es más difícil de mantener y actualizar debido al crecimiento de su código. 2. Si ocurre algún fallo, este afecta a toda la estructura. 3. Está limitado a la tecnología con la que se desarrolla y no puede usar otras. 	<ul style="list-style-type: none"> 1. Linux: para servidores. 2. Windows: aplicaciones de escritorio. 3. macOS: para el desarrollo de aplicaciones Web.
HÍBRIDA	<ul style="list-style-type: none"> 1. Facilidad a la hora de implementar entornos de nube. 2. Facilita la implementación de nuevas tecnologías sin tener que reescribir toda la aplicación. 3. Se pueden hacer pruebas con los módulos independientes lo que facilita la solución de problemas. 	<ul style="list-style-type: none"> 1. El despliegue de múltiples módulos se hacen más complicados y requiere de mayor coordinación. 2. La interacción entre módulos puede provocar sobrecarga en la red y problemas de comunicación. 3. Aunque los módulos sean independientes, pueden existir dependencias que complican las actualizaciones. 	<ul style="list-style-type: none"> 1. Linux: aplicaciones que requieren de mayor flexibilidad y adaptabilidad. 2. Windows: aplicaciones de escritorio y en entornos del servidor. 3. macOS: desarrollo de aplicaciones Web y móviles.
MICROKERNEL	<ul style="list-style-type: none"> 1. Al ser modular se adapta bien en entornos en la nube. 2. Facilita el escalado de componentes específicos sin afectar a toda la estructura. 3. Tiene mayor flexibilidad al poder añadir o eliminar módulos según la necesidad. 	<ul style="list-style-type: none"> 1. La comunicación entre el núcleo y los módulos externos puede afectar al rendimiento. 2. La compatibilidad entre las versiones del núcleo y los módulos puede ser complicada. 3. Esta arquitectura es inviable para aplicaciones simples. 	<ul style="list-style-type: none"> 1. Windows Fuchsia: utiliza un microkernel llamado "Zircon", que es más seguro. 2. Windows NT: es un sistema híbrido pero usa microkernel. 3. Android: Utiliza microkernel Linux.

ACTIVIDAD 3: Gestión de Procesos.

Completa la tabla que muestra la ejecución de procesos bajo el algoritmo Round Robin con un quantum de 2 Unidades de Tiempo (UT). Además, compara este algoritmo con otros más recientes, como el Multi-Level Feedback Queue (MLFQ), explicando su impacto en el rendimiento de la CPU:

Sabemos las siguientes características sobre un sistema operativo:

a) Utiliza el algoritmo de Round Robin, con quantum o rodaja de tiempo = 2 Unidades de Tiempo (UT).

b) Necesita ejecutar una serie de procesos, cuyos instantes de llegada y tiempos que tardan en ejecutarse se representan en la siguiente tabla:

PROCESO	INSTANTE DE LLEGADA	TIEMPO DE PROCESO
A	0	3
B	1	5
C	5	6
D	9	4
E	16	2

c) Los procesos de ejecutan en un sistema operativo ideal, es decir, en el que el sistema operativo no consume recursos de CPU.

d) Comenzamos a estudiar el sistema desde que entran nuestros procesos al sistema y considerando la Unidad de Tiempo 0 (UT0).

e) En la siguiente tabla se pueden apreciar los procesos que se ejecutan en estas condiciones desde la Unidad de Tiempo 0 (UT0) a la Unidad de Tiempo 2 (UT2).

PROCESO																				
E																				
D																				
C																				
B																				
A																				
Tiempo	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Respetando todas las restricciones dadas en el enunciado:

1. Completa la tabla anterior para las Unidades de Tiempo de la 3 a la 19, estableciendo el proceso que se ejecutará en cada Unidad de Tiempo e indicando los instantes de llegada de cada proceso.

2. Razona tu respuesta especificando el estado en el que se encuentra cada uno de los procesos para las unidades de tiempo de la 3 a la 7.

Dado que el algoritmo Round Robin utiliza dos unidades de tiempo en cada una de sus fases, desde la unidad de tiempo 3 nos situamos en el proceso “A” puesto que su tiempo de proceso será de un total de 3 (cuadros azules). Luego pasaremos al proceso “C” cuya unidad de tiempo es 5, volvemos a contar dos (cuadros azules) y pasamos de nuevo al proceso “B” para sumar otros 2 (cuadros azules).

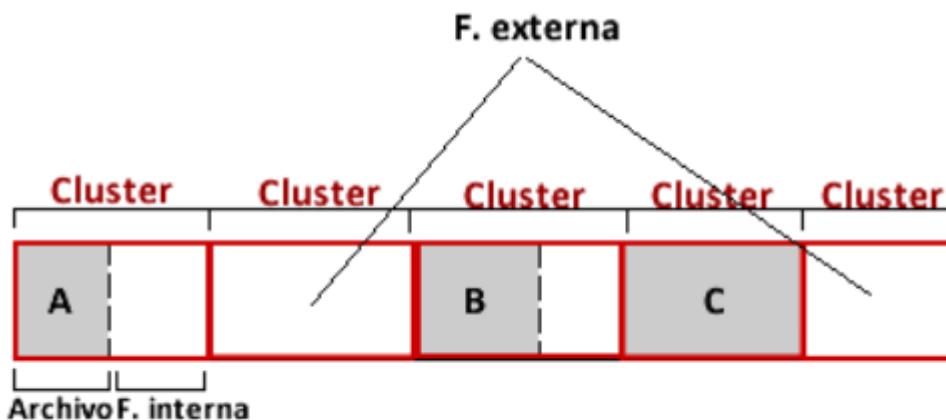
3. Realiza una breve comparativa entre el algoritmo Round Robin y el Multi-Level Feedback Queue (MLFQ), explicando en qué casos uno podría ser preferible al otro y cómo afecta al rendimiento de la CPU.

- **Algoritmo Round Robin:** (Algoritmo por turnos), se encarga de seleccionar todos los procesos de un grupo de forma equitativa, del primer elemento al último y repitiendo otra vez hasta terminar el proceso de computo.
Este algoritmo suele preferirse para el uso de juegos en línea para gestionar los tiempos de los jugadores entre las sesiones.
En cuanto al rendimiento de la CPU, al ser fijos sus procesos, los tiempos de respuesta son más predecibles. Pero puede verse afectado negativamente si no se gestionan bien los tiempos de los quantums.
- **Algoritmo MLFQ:** (Algoritmo Mediante Colas Multinivel Realimentadas), se dividen la cola de procesos en varias colas, una por cada tipo de trabajo y no permiten el movimiento de procesos entre distintas colas.
Este algoritmo es útil en entornos multitarea con usos de tiempo real, ya que puede gestionar las tareas sin retrasos.

En cuanto al rendimiento de la CPU es positivo en sistemas que usan una gran carga de procesos, sobre todo si hay muchos procesos que sean interactivos. Pero deberán estar bien optimizados para evitar posibles fallos.

ACTIVIDAD 4: Gestión de memoria:

- a) **Fragmentación Interna vs Externa:** Explica la diferencia entre fragmentación interna y externa. Usa ejemplos prácticos para ilustrar cada caso.
- **Fragmentación Interna:** un bloque de memoria es más grande que el espacio solicitado y se desperdicia el exceso de espacio.
EJEMPLO: Un sistema asigna memoria a procesos en bloques de 64 MB, en el que un proceso solo necesita 50 MB. Los 14 MB restantes en ese bloque se desperdiciarían.
 - **Fragmentación Externa:** hay espacios entre los bloques asignados y la memoria no puede usarse para solicitudes más grandes.
EJEMPLO: Un sistema de control asigna bloques de memoria a diferentes tareas. Si las tareas cortas han terminado y liberado memoria, los espacios que quedan pueden ser pequeños y no aptos para nuevas tareas que requieran de más espacio.



- b) **Memoria Virtual y Fragmentación:** Investiga si en un esquema de memoria virtual paginada puede existir algún tipo de fragmentación. Explica en qué circunstancias ocurriría y cómo se puede mitigar.

La fragmentación en los esquemas de memoria virtual paginada pueden darse en la fragmentación interna debido al tamaño fijo de las páginas y la fragmentación externa es menos común pero también puede darse en los sistemas combinados de paginación y segmentación. Para poder evitar estos problemas habría que elegir técnicas de gestión de memoria eficientes y elegir bien el tamaño de las páginas.

↓ ↓ ↓ ↓ ESQUEMA EXPLICATIVO DEBAJO ↓ ↓ ↓ ↓



Nota: utiliza diagramas o ejemplos prácticos para mejorar la comprensión de los conceptos.

ACTIVIDAD 5: Aplicaciones y tipos de licencias.

Indica una aplicación lo más reciente posible como ejemplo para cada uno de los siguientes tipos de licencias:

Tipo de Licencia	Aplicación	Fecha de Última Actualización
Software propietario	DISCORD	25/01/2023
Software libre	Google CHROME	22/10/2024
Software semilibre	Visual Studio Code	10/10/2024
Software de dominio público	LibreOffice	11/07/2024
Software con copyleft	WordPress	2/04/2024
Freeware	Skype	23/09/2024
Shareware (demo)	WinRAR	20/05/2024
Shareware (versión limitada por tiempo)	Norton Antivirus	24/06/2023