



UNIDAD 2: “S.O” Y SOFTWARE DE UN SISTEMA INFORMÁTICO.

DAM 1- SISTEMAS INFORMÁTICOS GRUPO A

INDICE

Actividad 1.....	pág.2
Actividad 2.....	pág.3
Actividad 3.....	pág.4
Actividad 4.....	pág.5
Actividad 5.....	pág.6-7
Actividad 6.....	pág.8-10
Actividad 7.....	pág.11-12

Actividad 1.- Tipos de aplicaciones informáticas.

Indica, justificando su uso, dos ejemplos de aplicaciones de propósito general y otros dos de propósito específico para cada uno de los siguientes entornos productivos:

- **Agencia de viajes:**

- Propósito general:**

- Para este caso en concreto, pueden ser útiles las consideradas como generadoras de presentaciones (para explicaciones y hojas de ruta de los viajes para los clientes) y las hojas de cálculo (para poder llevar un registro de los clientes y los viajes asociados).

- Propósito específico:**

- En esta ocasión pueden ser muy útiles las aplicaciones de contabilidad (para controlar los movimientos económicos) y las de gestión de redes (para controlar los archivos comunes entre las diferentes sucursales)

- **Agencia de publicidad y marketing:**

- Propósito general:**

- Para la agencia de publicidad y marketing, resultan esenciales las aplicaciones de gestión de texto (para la creación de campañas) y las aplicaciones destinadas para herramientas de comunicación (como las que se encargan de los servicios de mensajería para estar en contacto con sus clientes)

- Propósito específico:**

- Por último, van asociadas a este tipo de empresas, las aplicaciones de diseño gráfico y maquetación (como Photoshop, un imprescindible para realizar diseños) y las de administración de bases de datos (creando una base en la que se relacionan las campañas de los clientes, con sus tiempos, presupuesto, antigüedad, etc.)

Actividad 2.- Licencias software.

Indica para cada una de las siguientes aplicaciones, el tipo de licencia que utiliza, intentando dar en términos generales si se trata de una licencia de software con código abierto, con código cerrado o de dominio público (sin licencia), y de manera específica la licencia exacta que utiliza.

Por ejemplo: El programa de edición de gráficos vectoriales llamado Inkscape, usa una licencia GPLv3+, que es una licencia de software con código abierto, no permisiva, de tipo copyleft fuerte (software libre).

Las aplicaciones son las siguientes:

- **Microsoft Office 2021.**

Está dentro de las licencias de código cerrado, en el que el código fuente no está disponible, y es un software EULA (disponible para el usuario que lo ha comprado)

- **Google Chrome.**

Está dentro de las licencias de código abierto, el código fuente está disponible (casi en su totalidad, según indican en "chromium") es un software libre.

- **VLC Media Player.**

Está dentro de las licencias de código abierto, no permisivas de tipo copyleft fuerte. Es una licencia GPLv2.1+

- **Avast Free Antivirus.**

Es un software de código cerrado, cuyo código fuente no está disponible. Está dentro de la subcategoría de "Freeware" que tiene como característica la libre distribución, pero no su modificación.

- **Adobe Acrobat Pro.**

Por último, este software, de código cerrado, cuyo código fuente no está disponible para el usuario. Está en la categoría de los EULA (disponibles para los usuarios que lo han comprado). No es posible su distribución ni modificación.

Actividad 3.- Sistemas operativos: Última versión, requisitos hardware, licencia y campos de aplicación.

Realiza la siguiente tabla añadiendo una fila por sistema operativo con la última versión existente de:

- Windows 11 (canal de disponibilidad general).
- Ubuntu Desktop LTS (última versión con soporte de larga duración).
- iOS (para iOS, en lugar de "requisitos hardware", incluye una lista con los dispositivos que soportan la última versión).
- Android (para Android, incluye las características hardware de un smartphone específico que utilice la última versión, indicando de qué smartphone se trata).

Sistema Operativo	Última versión	Requisitos hardware			Licencia	Dispositivos en los que normalmente se instala
		Procesador	RAM	Espacio de almacenamiento		
Windows 11	22h2	1GHz (2 núcleos)	4GB	64GB	Retail	Ordenadores: Sobremesa, Portátil, Tablet
Ubuntu Desktop LTS	22.10	2 GHZ (2 núcleos)	4 GB	25GB	GPL	Ordenadores: Sobremesa, Portátil
iOS	16.1.1	<i>iPhone 8 y modelos posteriores, iPad Pro (todos los modelos), iPad Air 3.ª generación y modelos posteriores, iPad 5.ª generación y modelos posteriores, y iPad mini 5.ª generación y modelos posteriores compatibles)</i>			EULA	Tabletas y móviles
Android	13	Quad Core	2 GB	16GB	Freeware	Dispositivos móviles y tablets

Actividad 4.- Arquitectura interna de un sistema operativo.

Explica en qué consisten las siguientes arquitecturas de sistemas operativos:

- **Monolítica.**

Los sistemas que usaban arquitecturas monolíticas, eran los de primera generación. Consistían en Sistemas Operativos diseñados a medida. Todas las funciones se implementaban en el núcleo. Se trataba de un programa que tenía rutinas entrelazadas que se podían llamar entre sí. Son difíciles de mantener.

Ejemplo: MS-DOS

- **Microkernel.**

En este tipo de sistemas las tareas se distribuyen en pequeñas porciones de código modulares. Se intenta de esta manera aislar el núcleo de las operaciones de entrada/salida, sistema de archivos, etc. Es muy interesante de cara a la tolerancia de fallos, es seguro y es fácilmente portable entre diferentes tipos de hardware.

Ejemplo: MINIX 3

- **Híbrida.**

Una combinación entre las anteriores. Se dota al núcleo de la capacidad de gestionar ciertas funciones para lograr que su ejecución sea más rápida que si estuviera en el espacio del usuario.

Ejemplo: Microsoft Windows NT

(Saltar esta página, sección informativa)

Actividad 5.- Gestión de procesos.

Sabemos las siguientes características sobre un sistema operativo:

1. Utiliza el algoritmo de planificación de procesos **SJF** (el trabajo más corto primero), el cual es un algoritmo **no apropiativo**.
2. Necesita ejecutar una serie de procesos, cuyos instantes de llegada y tiempos que tardan en ejecutarse se representan en la siguiente tabla:

Proceso	Llegada	Tiempo de CPU
A	0	4
B	1	8
C	3	5
D	5	2
E	8	1

3. Los procesos se ejecutan en un sistema operativo ideal, es decir, en el que el sistema operativo no consume recursos de CPU.
4. Comenzamos a estudiar el sistema desde que entran nuestros procesos al sistema y considerando la Unidad de Tiempo 0 (UT0),
5. 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 1 (UT1).

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

Leyenda:

↓	Proceso ha llegado y está listo para ejecutarse en ese instante
F	Proceso termina en ese instante
#	Proceso se está ejecutando (# representa el instante de ejecución)
	Proceso está esperando en la cola de procesos listos

Respetando todas las restricciones dadas en el enunciado:

1. Completa la tabla anterior para las unidades de tiempo de la 2 a la 19, estableciendo el proceso que se ejecutará en cada unidad de tiempo e indicando los instantes de llegada de cada proceso. Utiliza la nomenclatura y/o simbología que aparece en la leyenda.

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

2. Razona tu respuesta especificando el estado en el que se encuentra cada uno de los procesos para las unidades de tiempo de la 2 a la 8.

Proceso A: Se está ejecutando y se encuentra en el instante 3 de su tiempo de ejecución.

Proceso B: Está esperando en la cola de procesos. Será el último ya que su tiempo de ejecución es el mayor y coincide en la cola con C.

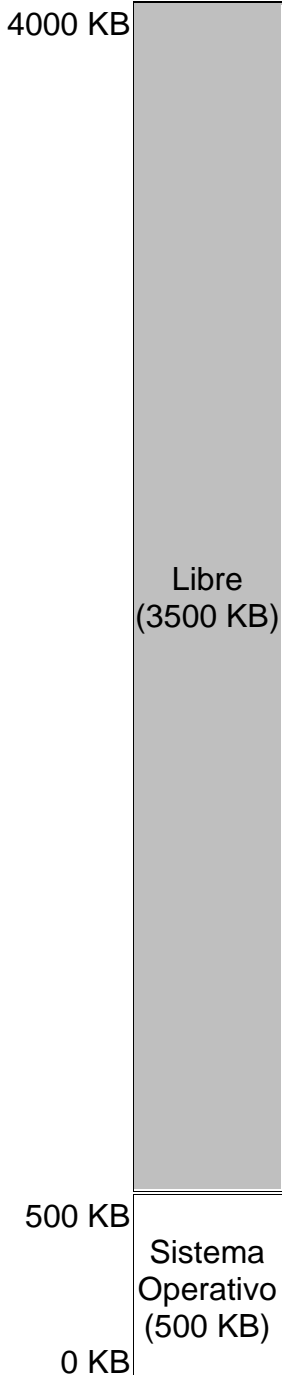
Proceso C: Comienza a ejecutarse en el instante 4, que es cuando termina el proceso A, y tiene preferencia respecto al B, por su menor duración.

Proceso D: Está esperando en la cola de procesos. Será el penúltimo, ya que el proceso E está a la par y tiene menor tiempo de ejecución.

Proceso E: en el instante 8, llega a la cola de procesos.

Actividad 6.- Gestión de memoria.

Supón un sistema en el que la gestión de memoria se realiza siguiendo un esquema de asignación de **particiones variables** en el que **no es posible realizar compactación** de memoria (no se pueden mover de sitio los procesos una vez hayan sido ubicados en memoria). La capacidad de la memoria es de 4000 KB, de los cuales 500 se encuentran ocupados por el sistema operativo, y el resto está disponible para ubicar los procesos que se ejecuten.



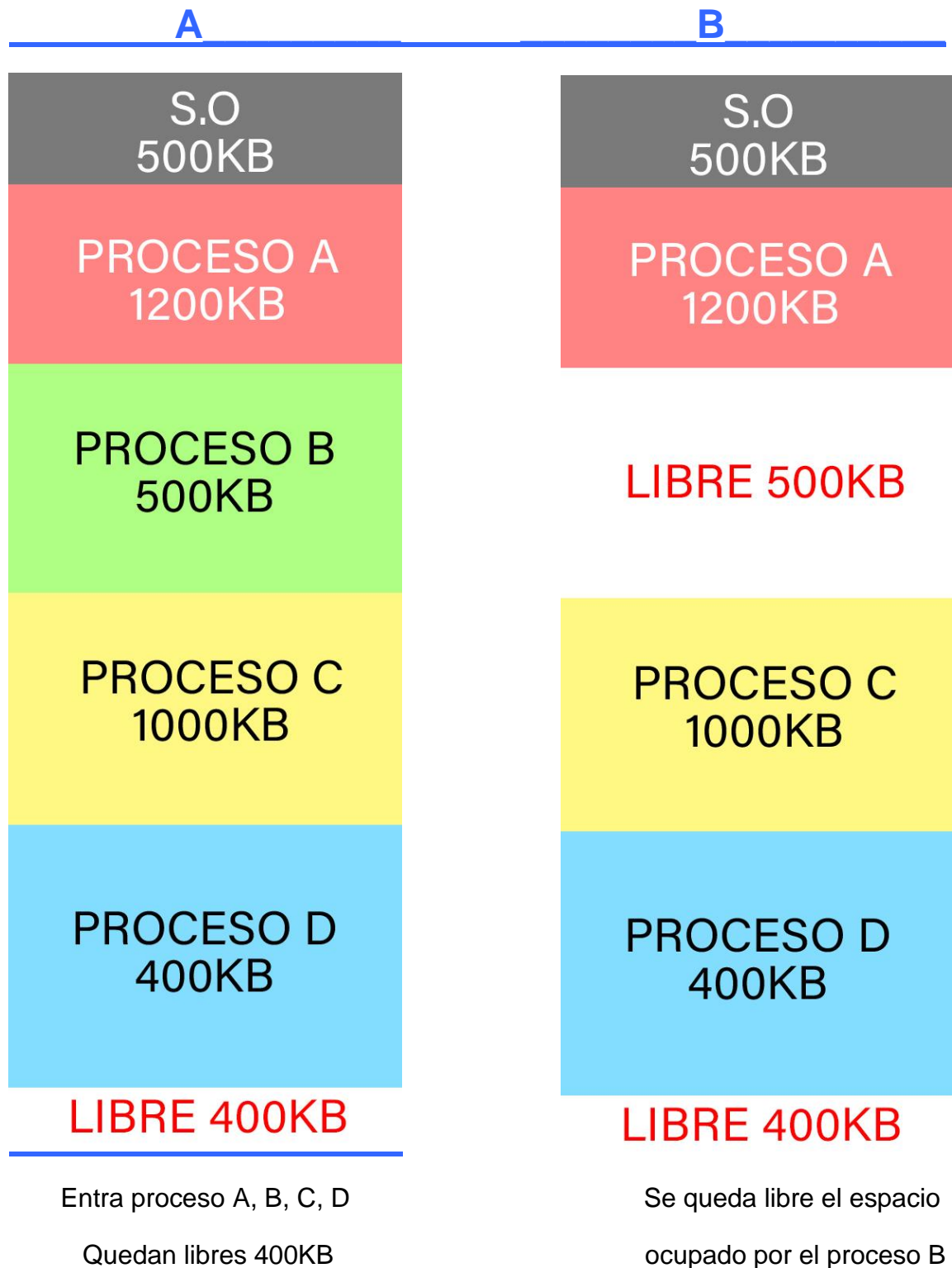
Se van produciendo las siguientes llegadas y salidas de procesos, por orden, que requieren la asignación o liberación de trozos de memoria:

1. Inicialmente solo está cargado el sistema operativo, sin ningún otro proceso (situación de partida).
2. Llega un proceso A de 1200 KB de tamaño y se intenta cargar en memoria.
3. Llega un proceso B de 500 KB de tamaño y se intenta cargar en memoria.
4. Llega un proceso C de 1000 KB de tamaño y se intenta cargar en memoria.
5. Llega un proceso D de 400 KB de tamaño y se intenta cargar en memoria.
6. El proceso B termina su ejecución y se libera el espacio que estaba ocupando en memoria.
7. Llega un proceso E de 600 KB de tamaño y se intenta cargar en memoria.

Realiza lo siguiente:

- a. Si consideramos que el dibujo de arriba se corresponde con el punto 1 de la lista anterior, incluye dos dibujos más en los que se muestre cómo se encontraría la memoria después de los instantes 5 (tras llegar D y cargarse en memoria) y 6 (tras terminar B y liberarse su espacio).
- b. Explica qué ocurre en el punto 7, cuando llega "E" e intenta cargarse en memoria. Comenta si ocurre algún tipo de fragmentación y, en caso afirmativo, qué tipo de fragmentación sería.

[Resolución en la siguiente página](#)





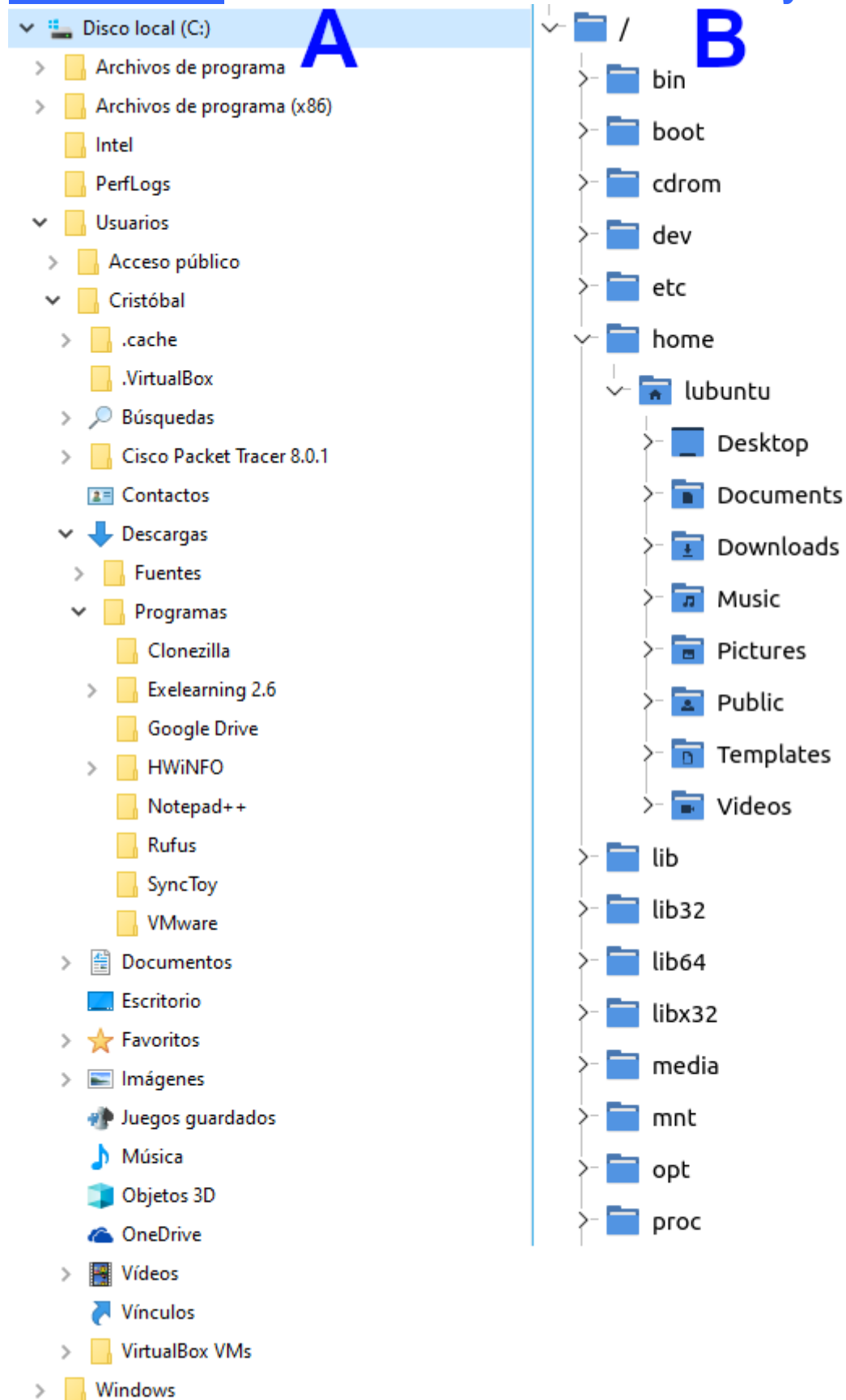
6.b (Instante 7)

Cuando el proceso "E" intenta cargarse solo hay dos espacios libres de 500KB y 400KB, y no puede entrar en la memoria.

Debido a la fragmentación externa, aunque la suma de los dos espacios libres es de 900kb que permitirían al proceso "E" cargarse, no es posible porque son compartimentos independientes que no se reasignan después de que el proceso haya sido eliminado de la memoria.

Se deduce que la fragmentación no es interna, debido a que, por definición, consiste en la diferencia de memoria usada y la que el S.O preveía que se iba a usar. Por tanto, en un esquema de particiones variables no es adecuado este tipo de fragmentación.

Actividad 7.- Estructuras de directorios y rutas.



Para el equipo con Windows (A), escribe las siguientes rutas:

- Una ruta absoluta al directorio "Fuentes".

c:\usuarios\cristobal\descargas\fuentes

- Una ruta relativa al directorio "Rufus", considerando que el directorio de trabajo/activo actual sea "Imágenes".

../cristobal/descargas/programas/rufus

- NOTA: Las rutas en Windows **no** son sensibles a mayúsculas y minúsculas.

Para el equipo con Linux (B), escribe las siguientes rutas:

- Una ruta absoluta al directorio "Downloads".

/home/lubuntu/Downloads

- Una ruta relativa al directorio "Pictures", considerando que el directorio de trabajo/activo actual sea "media".

../home/lubuntu/Pictures

- NOTA: Las rutas en Linux **sí** son sensibles a mayúsculas y minúsculas.