APELLIDOS		NOMBRE		Grupo
DNI		Firma		

- **No desgrape las hojas.**
- **Conteste exclusivamente en el espacio reservado para ello.**
- **Utilice letra clara y legible. Responda de forma breve y precisa.**
- **El examen consta de 9 cuestiones, cuya valoración se indica en cada una de ellas.**

---

1. Sea un sistema con gestión de memoria paginada, con páginas de 4KB, direcciones lógicas de 28 bits y 4GB de memoria física. En dicho sistema se está ejecutando un proceso del que se muestra **parte** del contenido de su tabla de páginas (en hexadecimal):

Página	Marco	Bv
0	521F5	1
1	112D2	1
2	43231	1
3		0
4	231F2	1
5	231F3	1
6	231F4	1
...	...	...

(1,0 puntos = 0,5 + 0,5)

1	**a)** Calcule, en hexadecimal, la dirección física que se genera si el proceso emite la dirección lógica 0002F1F (en hexadecimal). Exponga de forma concisa cómo la obtiene.
	b) Calcule, en hexadecimal, la dirección lógica de dicho proceso que genera un acceso a la dirección física 231F2288 (en hexadecimal). Exponga de forma concisa cómo la obtiene.

2. Un sistema operativo debe gestionar de forma eficiente la memoria de la máquina. Piense en la problemática que genera la gestión de memoria y responda:

(1,0 puntos = 0,5 + 0,5)

2	**a)** ¿Qué problema trata de resolver la técnica de compactación y en qué tipo o tipos de asignación contigua de memoria tienen utilidad su aplicación?
	b) ¿Qué utilidad tiene la MMU (Unidad de Manejador de Memoria) y qué problemas trata de resolver?

3. Considerando el mapa de memoria de un proceso UNIX, diga si las siguientes sentencias son verdaderas o falsas:

(0,6 puntos)

3	SENTENCIA	V/F
	La región de datos sin valor inicial del mapa de memoria que se genera al ejecutar un programa, es una región que no tiene soporte en ningún fichero.	
	La región de código con soporte en el archivo ejecutable del proceso tiene permisos de lectura, escritura y de ejecución.	
	Cuando un proceso mapea un archivo en memoria se crea una nueva región en el mapa de memoria de ese proceso a la que se accede mediante operaciones de entrada/salida.	
	La utilización de bibliotecas dinámicas se basa en la técnica de mapeo de las librerías en memoria durante la ejecución del proceso que las utiliza.	

4. Sea un sistema operativo que gestiona la memoria virtual mediante paginación con páginas de 4 KB. Los marcos libres se asignan por orden creciente de direcciones físicas utilizando políticas de reemplazo locales. A cierto proceso nuevo el sistema le asigna 3 marcos (del 0 al 2) y durante su ejecución se emiten las siguientes direcciones lógicas (en hexadecimal):

400000, 400014, 800000, 800024, 600030, 600034, 800280, 40060c, 200000, 400c10, 600f24

Indique qué páginas quedarán en la memòria física después del último acceso y cuál es la última dirección física a la que se accede en los siguientes casos. (Indique los marcos libres u ocupados con una página inválida con '—').

(2,0 puntos = 0,5 + 0,5 + 0,5 + 0,5)

4

a) Algoritmo de reemplazo LRU

marco	Página
0	
1	
2	

Última dirección física:

b) Algoritmo de reemplazo segunda oportunidad

marco	Página
0	
1	
2	

Última dirección física:

c) Algoritmo de reemplazo óptimo

marco	Página
0	
1	
2	

Última dirección física:

d) Suponga ahora que se aplica un algoritmo FIFO y que después del último acceso de la lista, el código invoca a la llamada *exec* de un ejecutable que realiza los siguientes accesos lógicos: 400000, 400010, 600020

marco	Página
0	
1	
2	

Última dirección física:

5. En un sistema con direcciones físicas y lógicas de 32 bits, se establece un mecanismo de paginación de dos niveles. Las páginas son de 4KB y las tablas de páginas de segundo nivel residen en memoria ocupando un único marco cada una.

(1,0 puntos = 0,5 + 0,5)

5	**a)** Determine el número de entradas que tendrán las tablas de páginas de segundo nivel. Considere que cada entrada de la tabla o descriptor de página utiliza 12 bits para los bits Bv, Br, Bm, R, W, X, etc.
	b) Describa la estructura de las direcciones lógicas y de las direcciones físicas de este sistema, así como el tamaño en bits de cada uno de los elementos que la componen.

6. El contenido de las tablas de descriptores de archivos abiertos correspondientes a tres procesos en un instante dado es el siguiente:

Tabla inicial	
0	STDIN
1	STDOUT
2	STDERR
3	
4	
5	

Tabla del proceso P1	
0	STDIN
1	STDOUT
2	STDERR
3	"result"
4	"fd_pipe[0]"
5	"fd_pipe[1]"

Tabla del proceso P2	
0	STDIN
1	"fd_pipe[1]"
2	STDERR
3	
4	
5	

Tabla del proceso P3	
0	"fd_pipe[0]"
1	"result"
2	STDERR
3	
4	
5	

Complete el fragmento de código en C con las primitivas POSIX necesarias para que al ejecutarlo se creen los tres procesos y el contenido de sus tablas de descriptores en los puntos marcado como /*** Tabla …….***/ sea el mostrado

(1.0 puntos)

6

```
int fd_pipe[2]; /*descriptor de tubo*/
int fd;     /* descriptor archivo regular**/

/*** Tabla Inicial ***/
  fd=open("result",O_WRONLY |O_CREAT|O_TRUNC,0666);


 /*** Tabla del proceso P1 ***/
  if (!(pid=fork())) {









 /***Tabla del proceso P2 ***/
    execlp("/bin/cat", "cat", "fich1", NULL);
  }
  if(!(pid=fork())){









  /***Tabla del proceso P3 ***/
    execlp("/usr/bin/wc", "wc", "-l",NULL);
  }
  close(fd_pipe[0]); close(fd_pipe[1]);
  close (fd);
  while(pid != wait(&status));
}
```

7. Dado el siguiente listado de un directorio en un sistema POSIX:

```
i-nodo   permisos   enlaces usuario  grupo tamaño     fecha       nombre
2021662 drwxr-xr-x   2      sterrasa  fso  4096       dec  9 2015  .
2021611 drwxr-xr-x   8      sterrasa  fso  4096       sep 10 2015  ..
2021663 -rwxr-sr-x   1      sterrasa  fso  1139706    dec  9 2015  copia
2021664 -rw-rw-r--   1      sterrasa  fso  9706       dec  9 2015  f1
2021665 -rw-rw-r--   2      sterrasa  fso  4157       dec  9 2015  f2
2021665 -r--r--rw-   2      sterrasa  fso  4157       dec  9 2015  f3
2021666 lrwxrwxrwx   1      sterrasa  fso     2       dec  9 2015  f4->f1
```

(1.25 puntos = 0.75 + 0.5)

7

a) Teniendo en cuenta que el programa `copia` copia el contenido del archivo que recibe como primer argumento en otro cuyo nombre se indica como segundo argumento. Es decir, "`copia a b`" copia el contenido del archivo `a` al archivo `b`, si `b` no existe lo debe crear y después realizar la copia. Rellene la tabla, indicando en caso de éxito cuáles son los permisos que se van comprobando y, en caso de error, cuál es el permiso que falla y por qué.

(UID,GUI)	ORDEN	¿FUNCIONA?	JUSTIFICACIÓN
(pepe, fso)	copia f1 f5		
(sterrasa, fso)	copia f1 f3		
(ana, etc)	copia f3 f4		

b) Indique qué valores de la columna *enlaces* de listado anterior se verían afectados si se elimina la entrada de directorio f3 (`rm f3`). ¿Y si se elimina la entrada f4?. Justifique sus respuestas

8. Una partición de 32MBytes, de los cuales 2Mbytes están ocupados con estructuras propias del sistema de archivos, está organizada en bloques de 512 bytes, el puntero a bloque es de 32 bits. Calcule:

(1,0 puntos = 0.5 + 0.5)

8	**a)** Tamaño máximo de un archivo si se utiliza asignación indexada simple, con un solo bloque de índices por archivo
	b) Tamaño máximo de un archivo si se utiliza asignación enlazada

9. Un disco con una capacidad de 6GB, se formatea con una versión de MINIX, cuyos tamaños son los siguientes:

- El bloque de arranque y el superbloque ocupan 1 bloque cada uno.
- El tamaño del nodo-i es de 32 bytes (7 punteros directos, 1 indirecto,1 doble indirecto).
- Con punteros a zona de 16 bits.
- Cada entrada de directorio ocupa 16 bytes.
- 1 zona = 1 bloque = 1 KByte
- Al formatear se ha reservado espacio en la cabecera para para 4.096 nodos-i

El esquema de los diferentes elementos del disco es el siguiente:

Arranque	Superbloque	Mapa de bits Nodos-i	Mapa de bits Zonas	Nodos- i	Zonas de datos

Se pide:

a) Calcule el número de bloques que ocupa el Mapa de bits nodos-i, el Mapa de bits Zonas y los Nodos-i.

b) El contenido de los directorios es el que se muestra, suponiendo que los archivos regulares ocupan 10 KBytes cada uno ¿cuántos i-nodos están ocupados?¿cuántas zonas de datos están ocupadas?

1	.
1	..
4	usr
10	bin

4	.
1	..
20	list
12	alumno

12	.
4	..
26	prac

10	.
1	..
35	calc

(1,15 puntos = 0,5 + 0,65)

9	a)
	b)