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 8 cuestiones, cuya valoración se indica en cada una de ellas.**

---

1. Sea un sistema que gestiona una memoria de 1200 KBytes con asignación contigua con particiones variables. La política de asignación de huecos es Worst Fit (peor hueco) con compactación. El algoritmo de compactación minimiza desplazamientos en memoria hasta encontrar un hueco suficiente para ubicar el proceso que solicita memoria en cada instante, optando siempre por desplazar procesos hacia las direcciones más altas. Considere la ocupación que se muestra (Estado Inicial) en la siguiente tabla:

0 1200KB-1

SO 80KB	P1 160KB	HUECO 400KB	P2 120KB	HUECO 440KB

(1,0 puntos = 0,6 +0,4)

a)Gestione la memoria, aplicando compactación si es necesario, e indique la dirección base asignada a cada proceso al producirse de manera secuencial, una tras otra, las nuevas solicitudes descritas a continuación:

Dirección Base	Estado Inicial	Llega P3 (280K)	Llega P4 (200K)	Llega P5 (280K)	Llega P6 (80K)
Base de P1					
Base de P2					
Base de P3	----				
Base de P4	----				
Base de P5	----				
Base de P6	----				

b) Indique de forma justificada qué tipo de fragmentación aparece en particiones variables y si se da en algún momento del apartado a) estime la cantidad

2. Respecto a la compartición de páginas en un sistema con paginación, indique si son verdaderas (V) o falsas (F) cada una de las siguientes afirmaciones: (Nota: Un error penaliza una respuesta correcta)

(0,8 puntos)

2	SENTENCIA	V/F
	Existen descriptores de página pertenecientes a procesos diferentes que contienen el mismo número de marco los cuáles corresponde a páginas compartidas	
	La técnica de copy-on-write, permite compartir páginas con permisos de escritura	
	Los procesos que comparten páginas ubicadas en memoria, comparten también la Tabla de páginas	
	Una página cargada en memoria con permisos de su región r-xp puede ser compartida por varios procesos	
	La probabilidad de llegar a una situación de hiperpaginación se incrementa al aumentar el número de páginas compartidas entre procesos	

3. Sea un sistema con direcciones físicas y lógicas de 24 bits y páginas de 4KB que gestiona la memoria virtual mediante paginación. Los marcos libres se asignan en orden creciente y utiliza un algoritmo de reemplazo LRU con ámbito LOCAL. **(1.2 puntos = 0,3 + 0,9)**

3

a) Calcule la serie de referencias para la siguiente secuencia de direcciones lógicas (en hexadecimal) emitidas por los procesos A y B durante su ejecución: A:40000, A:40014, B:80000, B:80024, B:60030, A:60034, B:80280, B:4060c, A:20000, A:40c10, B:60f24

b) Teniendo en cuenta que el sistema asigna los marcos 0 y 1 al proceso A y los marcos 2, 3, 4 al proceso B y que inicialmente dichos marcos están vacíos. Complete la siguiente tabla con la evolución del contenido de la memoria principal para la serie de referencias obtenida en el apartado anterior e indique el número de fallos de página que produce

Marco									
0 (A)									
1 (A)									
2 (B)									
3 (B)									
4 (B)									

Número de Fallos de página:

4. Sea un sistema con paginación por demanda, dos niveles de paginación con 16 descriptores de página de primer nivel, direcciones lógicas y físicas de 24 bits, y páginas de 4Kbytes. Dicho sistema asigna 3 marcos a cada proceso y está ejecutando el proceso A. En el instante t=10, el proceso A tiene ocupado los marcos 0xf00, 0xf01 y 0xf02, por las páginas 0x001, 0xf2f y 0x11a respectivamente.

(1,6 puntos = 0,4 + 0,4 + 0,8)

4

a) Describa la estructura de las direcciones lógicas y direcciones físicas de este sistema, así como el tamaño en bits de cada uno de los elementos que la componen.

b) Indique razonadamente el contenido de los descriptores de página con el bit de validez a válido (v=1) para el proceso A en el instante t=10.

c) Complete la siguiente tabla indicando de forma razonada las correspondientes direcciones lógicas o físicas, que podrían haber sido emitidas o accedidas para el proceso A en el instante T=10

Direc. Lógica	Dirección física	¿Es posible?, Justificación
	0xf01f02	
	0xf02f01	
0x11a13b		
0x13b11a		

5. Complete el programa en código en C del apartado a) con las primitivas POSIX necesarias (una en cada línea con número subrayado) para que un proceso padre lea el contenido del archivo "message.txt" y envíe 2 copias concatenadas de este a su proceso hijo a través de un tubo. El proceso hijo reenviará el contenido que reciba de este tubo a la salida estándar mediante el programa cat (Nota: la orden cat sin argumentos escribe en la salida estándar todo lo que lee de la entrada estándar). **(1,4 puntos= 0,8 + 0,6)**

5 **a)**

```
1  #include <all_needed>
2  #define SIZE 80
3  #define MODE O_RDONLY
4  int main( int argc, char **argv ){
5    int fd, fdp[2], readbytes, ncopy;
6    char buffer[SIZE];
7
8    if (fork() == 0){ // child
9
10       close(fdp[0]);
11
12       execlp("cat","cat", NULL);
13   }
14   else { // parent
15       close(fdp[0]);
16
17       for(ncopy=0; ncopy<2; ncopy++){
18         while((readbytes= read(fd, buffer, SIZE)) > 0){
19           write(                        ); /*complete*/
20         }
21         lseek(            ); /*complete*/
22       }
23
24
25   }
26   wait(NULL);
27   exit(0);
28 }
```

b) Rellene la tabla de descriptores de archivos del proceso hijo, en el momento en que se ejecuta la línea 12. Haga lo mismo para el proceso padre en la línea 17. Las tablas deben ser correctas para cumplir con los requisitos y la implementación de la sección a)

Tabla del Poceso. hijo en 12	
0	
1	
2	
3	
4	
5	

Tabla del Proceso. padre en 17	
0	
1	
2	
3	
4	
5	

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

```
 i-nodo   permisos  enlaces usuario  grupo tamaño     fecha     nombre
37093377 drwxr-xr-x  3       marta   disca   4096 dic 11 11:57 .
32448485 drwxr-xr-x  3       marta   disca   4096 dic 11 12:02 ..
32448767 -rwsr-xr-x  1       marta   disca 141528 dic 11 10:31 cp2
33373385 dr-xrwxr-x  2       marta   disca   4096 dic 11 12:02 dir1
32448804 lrwxrwxrwx  1       marta   disca      4 dic 11 10:35 dir2 -> dir1
32448793 -r--r--r--  1       marta   disca    337 dic 11 10:33 f1
32448802 -rw-r--r--  3       marta   disca    402 dic 11 10:33 f2
32448802 -rw-r--r--  3       marta   disca    402 dic 11 10:33 f3
32448803 lrwxrwxrwx  1       marta   disca      2 dic 11 10:34 f4 -> f3
```

(1,4 puntos = 0,8 + 0,6)

6 **a)** El programa `cp2` es similar a la orden cp de linux. Cuando el último parámetro de cp2 es un directorio copia los archivos indicados en dicho directorio (creándose si no existen). Asuma que el directorio dir1 está vacío, conteniendo sólo las entrada "." y "..", e indique 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,GID)	ORDEN	¿FUNCIONA?	JUSTIFICACIÓN
(pepe, disca)	cp2 f1 dir1		
(juan, fso)	cp2 f1 dir2		
(ana, alum)	cp2 f1 f2		

b) Justifique el valor de la columna enlaces de las siguientes entradas, indicando cuál es o deduciendo cuál puede ser la razón para que dicha entrada tenga ese número de enlaces

Nombre	Enlaces	Justifique
.	3	
dir1	2	
f2	3	

7. Un sistema de archivos está organizado en bloques de 512 Bytes, con punteros a bloque de 16 bits. En dicho sistema, existe un archivo "*ejemplo.txt*" de 32KBytes. Para cada uno de los métodos de asignación indicados, calcule el número de accesos a bloques de datos necesarios para leer el Byte 16900 del archivo *ejemplo.txt*. Justifique su respuesta. **(1,0 puntos = 0,5 + 0,5)**

7	**a)** Asignación Enlazada
	b) Asignación indexada

8. Un disco con capacidad de 64 MBytes, se formatea con una versión de MINIX, cuyas especificaciones son las siguientes:

- El bloque de arranque y el superbloque ocupan 1 bloque cada uno
- Tamaño del nodo-i de 32 Bytes (7 punteros directos, 1 indirecto,1 doble indirecto).
- Punteros a zona de 16 bits
- Entrada de directorio de 16 Bytes: 2 Bytes para nodo-i, 14 Bytes para el nombre
- 1 zona = 1 bloque = 1 KByte

(1,6 puntos = 0,3 + 0,8 + 0,5)

8	**a)** Calcule el número máximo de nodos-i que puede tener este sistema. Justifique su respuesta.

b) Suponga que el disco se formatea reservando espacio en la cabecera para un total de 32768 nodos-i (32 K nodos-i). Calcule el número de bloques que ocupa cada uno de los elementos de la cabecera y el área de datos. Justifique su respuesta.

Arranque	Super bloque	Mapa de bits de Nodos-i	Mapa de bits de Zonas	Nodos- i	Zonas de datos

c) Suponga que, en un momento dado, hay un total 32768 nodos-i (32K nodos-i) ocupados en el disco con un único directorio, el directorio raíz, y archivos regulares todos de 1KByte de tamaño. Indique de forma justificada el número de zonas del área de datos que estarán ocupadas en dicha partición.