



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, en cada una de ellas se indica su puntuación.**

1. Sea un sistema con una memoria principal de 2Mbytes gestionada con particiones de tamaño variable y que selecciona procesos de la cola de entrada con criterio FIFO para ubicarlos en memoria. Al iniciar el sistema se carga el SO (150 KB) en las posiciones más bajas de memoria. A continuación se solicita la ejecución de 4 procesos, en el siguiente orden: P1(200KB), P2(450KB), P3(150KB), P4(250KB) y P5(650KB), entre paréntesis se indican los requerimientos de memoria. Después de finalizar P2 y P4 llegan los procesos P6(250KB), P7(150KB) y P8(300KB), en dicho orden.

- Utilice una política Best fit (mejor hueco), e indique el estado de la memoria, ocupación y huecos, después de asignarle memoria a P8.
- Utilice una política Worst fit (peor hueco), e indique el estado de la memoria, ocupación y huecos, después de asignarle memoria a P8.

(1.0 puntos)

1	<i>a) Best fit</i>
	<i>b) Worst fit</i>

2. Un sistema dispone de 2 GB de espacio de direccionamiento lógico, páginas de 2 KB y 1 GB de memoria física.

- Calcule la dirección física correspondiente a la dirección lógica 4119, para una gestión de memoria mediante paginación. Las primeras entradas de la tabla de páginas se muestra a continuación. Justifique la respuesta.



Tabla de páginas

Página	Marco	Bit de validez
0	27	v
1	8	v
2	500	v
3	0	v

- b) Calcule la dirección lógica de un proceso que genera la dirección física 10288, en un esquema de segmentación paginada con 64 segmentos máximo por proceso. Las tablas de páginas para los segmentos 0, 1 y 2, de dicho proceso son:

Tabla de Páginas Segmento 0		
Página	Marco	Bit Val.
0	2	v
1	800	v
2	1024	v
3	3	v

Tabla de Páginas Segmento 1		
Página	Marco	Bit Val.
0	8	v
1	0	v
2	328	v
3	5	v

Tabla de Páginas Segmento 2		
Página	Marco	Bit val.
0	500	v
1	21	v
2	1	v
3	82	v

(1.25puntos)

2	a)
	b)

3. El sistema operativo de cierto computador gestiona la memoria virtual mediante paginación con páginas de 4 KB. Se asignan marcos libres, si los hay, por orden creciente de direcciones físicas. En un momento dado, el sistema ha de repartir 3 marcos (del 0 al 2), que inicialmente se encuentran libres, entre dos procesos nuevos A y B.

- Aplicando un algoritmo de reemplazo LRU con reemplazo global, indique la evolución del contenido de la memoria física si se hace referencia a las siguientes páginas: A0, B0, A1, A0, A2, A1, B0, A3, B3. Diga el número de fallos de página que se producen
- Aplicando un algoritmo de reemplazo de Segunda Oportunidad con reemplazo global, indique la evolución del contenido de la memoria física si se hace referencia a las siguientes páginas: A0, B0, A1, A0, A2, A1, B0, A3, B3. Y el número de fallos de página que se producen.
- Justifique si el algoritmo de reemplazo LRU presenta o no la anomalía de Belady.

Justifique todas las respuestas



(2,0 puntos)

3

a) LRU con reemplazo global

marco

0									
1									
2									

b) Segunda Oportunidad con reemplazo global

marco

0									
1									
2									

c) Justifique si el algoritmo de reemplazo LRU presenta la anomalía de Belady

4. En un sistema de memoria con paginación a dos niveles, la tabla de páginas del primer nivel puede albergar hasta 4096 referencias a tablas de páginas del segundo nivel. El tamaño de página es de 32 KBytes. El espacio de direcciones lógicas de este sistema es de 8GBytes y el espacio de direcciones físicas de 1GBytes,
- a) Describa la estructura de las direcciones lógicas y de las direcciones físicas de este sistema.
- b) Indique el número de descriptores de páginas que puede contener cada una de las tablas de páginas de segundo nivel.

(0,75)

4

a)

b)



5. Analice el siguiente programa denominado *ejtubo2.c*, y suponiendo que no se producen errores al ejecutar las llamadas, indique de forma justificada:

- El contenido de las tablas de descriptores en el punto donde está los comentarios */**(1) tabla(s) de descriptores */* y */**(2) tabla(s) de descriptores */* para cada uno de los procesos que alcancen durante su ejecución dicha la línea.
- Realice un diagrama del esquema de comunicación que llevan a cabo los procesos al ejecutarlo.
- Indique de forma justificada lo que se muestra por pantalla al ejecutar dicho código.

```
#include <string.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>

int main(int argc, char *argv[]) {
    int i, fd[2];
    pipe(fd);
    for(i=0; i<2; i++) {
        if (fork() == 0) {
            dup2 (fd[0], STDIN_FILENO);
            /**(1) tabla(s) de descriptores */
            close (fd[0]);
            close (fd[1]);

            execlp("/bin/cat", "cat", NULL);
            fprintf(stderr, "The exec of %s failed", argv[1]);
            exit(1);
        }
    }
    dup2 (fd[1], STDOUT_FILENO);
    close (fd[0]);
    close (fd[1]);
    /**(2) tabla(s) de descriptores */
    execlp("/bin/ls", "ls", "-l", NULL);
    perror("The exec of ls failed");
    exit(1);
}
```

(1,0 puntos)

5	a)
	b)
	c)

6. El siguiente programa *ejtubo1.c* crea dos procesos que se comunican con un tubo. Complete el programa con llamadas e instrucciones en los lugares indicados con **/**ajustad descriptores (X)**** y sustituya los comentarios **/**descrX**** por las variables adecuadas de manera que:

- El proceso padre le transmite al hijo, por el tubo, todo lo que lee por su entrada estándar.

- El proceso hijo imprima por su salida estándar todo lo que le recibe por el tubo.

Además ambos procesos cuentan el número de caracteres que leen o escriben en el tubo y escriben un mensaje en su salida estándar con esta información.

Ejemplo de ejecución: `$ echo "prueba" | ./ejtubo1`

*prueba
hemos escrito 7 caracteres
hemos leído 7 caracteres*

```
#include <string.h>
#include <stdio.h>
int main(void){
    int tubo[2];
    char letra;
    int leo=0, escribo=0;
    pipe(tubo);
    if (fork()==0){
        /* hijo lee del tubo y escribe en salida estándar */
        /**ajustad descriptores (1)**
        while (read(/**descr1** , &letra, sizeof(letra))==sizeof(letra)){
            write(/**descr2** , &letra, sizeof(letra));
            escribo++;
        }
        /**ajustad descriptores (2)**
        printf("\nhemos escrito %d caracteres\n", escribo);
    }else{
        /* padre lee de la entrada estándar y lo escribe en el tubo */
        /**ajustad descriptores (3)**

        while (read(/**descr3** , &letra, sizeof(letra))==sizeof(letra)){
            write(/**descr4** , &letra, sizeof(letra));
            leo++;
        }
        /**ajustad descriptores (4)**
        printf("\nhemos leído %d caracteres\n", leo);
    }
}
```

(1,0 puntos)

6



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

```
drwxr-xr-x      2 sterrasa  fso          4096 sep  8   2012 .
drwxr-xr-x     11 sterrasa  fso          4096 dec 10  14:39 ..
-rwsrw-r-x      1 sterrasa  fso        1139706 sep  9   2012 copia
-rw-rw-r--      1 sterrasa  fso         634310 sep  9   2012 f1
-r--r--rw-      1 sterrasa  fso         104157 sep  9   2012 f3
```

Donde el programa `copia` copia el contenido del archivo que recibe como primer argumento en otro cuyo nombre es el nombre recibido como segundo argumento. Rellene la tabla, indicando en caso de éxito cuales son los permisos que se van comprobando y, en caso de error, cuál es el permiso que falla y porqué.

(1,0 puntos)

7					
	Usuario	Grupo	Orden	¿Funciona?	Justificación
	Inma	fso	copia f1 f2		
	Sara	ltp	copia f1 f2		
	Vicent	tal	copia f1 f3		

8. Dado el siguiente listado de un directorio en un sistema POSIX, obtenido después de un ejecutar la orden: `$ls -lia:`

```
Nodo-i    permisos  enlaces  uid      gid      size
9176729  drwxr-xr-x  2  silterb1  disca-upvnet  4096  2012-12-11 16:00 .
7212687  drwxr-xr-x  5  silterb1  disca-upvnet  4096  2012-12-11 15:58 ..
9176730  -rw-r--r--  1  silterb1  disca-upvnet   25  2012-12-11 15:59 f1
9176732  lrwxrwxrwx  1  silterb1  disca-upvnet   2  2012-12-11 15:59 f2 -> f1
9176733  -rw-r--r--  2  silterb1  disca-upvnet   32  2012-12-11 16:00 f3
9176733  -rw-r--r--  2  silterb1  disca-upvnet   32  2012-12-11 16:00 f4
```

Indique de forma justificada:

- El número de archivos diferentes a los que se hace referencia con las entradas de este directorio
- El numero de enlaces del archivo “.”

(0,75 puntos)

8	a)
---	----



b)

9. Un disco con una capacidad de 8GB, 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 64 bytes, con punteros a zona de 32 bits (7 punteros directos, 1 indirecto, 1 doble indirecto) .
- Cada entrada de directorio ocupa 32 bytes.
- 1 zona = 1 bloque = 2Kbytes
- 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	Super bloque	Mapa de bits Nodos-i	Mapa de bits Zonas	Nodos- i	Zonas de datos
----------	-----------------	-------------------------	-----------------------	----------	----------------

Se pide:

- Calcule el número de bloques que ocupa el Mapa de bits nodos-i, el Mapa de bits Zonas y los Nodos-i, así como el número de Zonas de datos.
- Suponga que en este disco existe únicamente el directorio raíz que contiene 10 archivos regulares, cada uno de ellos de 50KBytes e indique de forma justificada el número de zonas de datos ocupadas para este caso.

(1,25 puntos)

9 a)

b)