

	EXAMEN FINAL SISTEMAS OPERATIVOS	3 de febrero de 2013
	Nombre y Apellidos: _____	NIF: _____.

1. (1pt) Considere el siguiente:

```
char buf1[5] = "XXXX";
char buf2[5] = "AAAA";
int main() {
    int fd1,fd2;
    fd1=open("prueba",O_RDWR | O_CREAT | O_TRUNC, 0666 );
    if ( fork() == 0) {
        fd2=open("prueba",O_RDWR );
        write(fd1,buf1,4);
        write(fd2,buf2,4);
        lseek(fd1,0,SEEK_SET);
        read(fd1,buf1,4);
        close(fd2); close(fd1);
    }
    else {
        strcpy(buf1, "BBBB");
        wait(NULL);
        write(fd1,buf2,4);
        read(fd1,buf1,4);
        close(fd1);
    }
}
```

- a) (0.5pts) ¿Cuál será el contenido del fichero *prueba* al finalizar la ejecución? y ¿Qué contendrá la variable "buf1" tanto al finalizar el proceso hijo como al finalizar el padre? Justifique su respuesta
- b) (0.5pts) Indique si la ejecución del código generará algún error o si hay varias posibilidades en el resultado de la ejecución.

2. (2pts) En un sistema con memoria virtual paginada se hacen las siguientes referencias:

a, d, c, a, b, d, f, a, e, f, a, e, d, f, a, g

- c) (1pt) Suponiendo una memoria física de cuatro marcos de página inicialmente vacíos, indique el número total de fallos de página tras aplicar los algoritmos: **óptimo, FIFO, reloj y LRU**.
- d) (1pt) Indicar qué sucede y cuáles de las acciones son realizadas por el sistema operativo (especificando a qué estructuras de datos accede y cómo las modifica) en los siguiente casos:
- i) un proceso intenta escribir en una página de sólo lectura;
 - ii) un proceso intenta acceder a una dirección virtual de su espacio de direcciones correspondiente a una página que no está en memoria.

3. (2pts) Supongamos un disco duro con 512 cilindros, 2 cabezas y 64 sectores de 1KB por pista. El disco gira a 3000rpm y tiene un mecanismo de cabezas móviles con un tiempo de búsqueda de 0.5ms. Este disco duro se formatea empleando un sistema de ficheros tipo FAT-16, reservando para la tabla FAT las posiciones Cilindro-Pista-Sector mas bajas y colocando el directorio raíz a continuación (el directorio raíz ocupa 1 bloque).

Con el disco recién formateado, se crea un fichero nuevo en el directorio raíz y se escriben 10bytes en él, seguidamente se adelanta en 5000 bytes el apuntador de posición del fichero y se escriben otros 100 bytes. Se pide:

- a. (0.5pts) ¿En qué dirección CPS comienza el directorio raíz? ¿Y el fichero?
- b. (1.5pts) Si el fichero se cierra y en un momento posterior se quieren leer los 100 últimos bytes, calcular tiempo necesario para realizar la operación para el caso mejor y para el caso peor. Asumir que no hay ningún dato relacionado con el sistema de ficheros en memoria RAM y que inicialmente estamos en (0,0,0).

4. (2pts): El sistema de ficheros de un SO diseñado a partir de UNIX utiliza bloques de disco de 4096 bytes de capacidad. Para el direccionamiento de estos bloques se utilizan punteros de 32 bits. Cada nodo-i tiene 8 punteros de direccionamiento directo, 1 puntero indirecto simple y 1 puntero indirecto doble.

- a. (0.75pts) ¿Cuál será el tamaño máximo de un fichero suponiendo despreciable el espacio ocupado por el superbloque, el mapa de bits y la tabla de nodos-i? Si se modifica el tamaño de puntero pasándolo a 64 bits, ¿cuál será el nuevo tamaño máximo?
- b. (0.75pts) Supongamos que el sistema de ficheros contiene la información de las tablas de abajo (sólo se muestra el primer puntero directo). Dibuje el árbol del directorio empleando óvalos para los directorios y rectángulos para los ficheros. Complete el campo *Enlaces* en la tabla de nodos-i.

Tabla de nodos-i

nodo-i	1	2	3	4	6	7	9	10	12
Enlaces	NA	NA	NA	___	NA	NA	___	___	___
Tipo F/D	D	D	D	F	D	D	F	F	F
Directo	4	5	6	13	8	9	14	12	15

Lista de bloques:

Bloque 4		Bloque 5		Bloque 6		Bloque 8		Bloque 9	
.	1	.	2	.	3	.	6	.	7
..	1	..	1	..	1	..	2	..	3
A1	2	A5	6	A6	7	a8	9	a10	10
A2	3					a9	10	a11	12
a3	4								

5. (3pts): Para agilizar el paso por caja, cierto supermercado ha decidido implantar un sistema de cola único. Este sistema consiste en que los clientes que deseen pasar por caja esperen en una única cola, repartiéndose por las cajas libres en orden de llegada. El funcionamiento es el siguiente:

1. La caja libre se anuncia a través de un cartel luminoso.
2. Al llegar un nuevo cliente éste se coloca en la cola común a la espera de ser el primero y de que haya una caja libre anunciada en el cartel. Cuando esto ocurre, borra el cartel y se dirige a dicha caja, poniendo los artículos en la cinta.
3. En caso de que haya más de una caja libre, sólo un cajero deberá poner el número de su caja en el cartel, el resto de cajas libres esperaran a que el cartel esté vacío.

a) (2pts) Escriba un programa que sincronice a los cajeros y a los clientes, empleando como mecanismo únicamente **semáforos**, de acuerdo con la estructura siguiente:

```
// Variables compartidas
int cajaLibre=-1
```

<pre>thread_Caja(int n) { while(1) { //Espero a que el cartel esté vacío cajaLibre=n; //Avisar a un cliente atenderCaja(n); } }</pre>	<pre>thread_Cliente(){ // Esperar turno en la cola única // Guardar el número de la caja nCaja=cajaLibre; // Marcar el cartel como libre y avisar ponerArticulos(nCaja); pthread_exit(NULL); }</pre>
---	--

Nota: Asuma que en el proceso de compra, esto es una vez que un cajero está atendiendo mediante `atenderCaja(n)` y un cliente está poniendo artículos en la cinta correspondiente invocando `ponerArticulos(n)`, hay sincronización implícita entre ellos.

b) (1pt) Los requisitos que debe ofrecer cualquier solución para resolver adecuadamente el problema de la sección crítica son: exclusión mutua, progreso y espera limitada. Explique brevemente en qué consiste cada uno de los tres.