APELLIDOS:	**DNI:**
NOMBRE:	
FILA:	**COLUMNA:**

Sistemas Operativos – Facultat d´Informàtica de Barcelona - UPC

Fecha: 19 de Enero de 2007 **Duración: 3 horas**

Notas: Las notas se publicarán el **Viernes 26 de Enero** en el RACÓ a las 18:00 horas.

Revisión: La fecha, hora y lugar exactos de la revisión del examen se publicarán junto con las notas. Estad atentos.

ATENCIÓN: Las preguntas se tienen que contestar en las mismas hojas de examen utilizando el espacio reservado a tal efecto. Asegúrate de poner NOMBRE y APELLIDOS, DNI, fila y columna en cada una de las hojas. El examen se tiene que entregar en bolígrafo negro o azul. Se pueden utilizar el documento con las llamadas al sistema.

Exercici 1: (2 punts)

Respon **breument** a les següents preguntes:

a) Què fa un planificador a mig termini? Quan s'executa?

b) Perquè és necessari traduir les adreces lògiques a físiques?

c) Diferències entre dispositiu lògic i dispositiu virtual.

d) Diferències entre una llibreria de sistema i una de llenguatge.

Exercici 2: (2 punts)

Tenim un ordinador amb un sol processador, on corre un sistema tipus UNIX amb memòria virtual i paginació. Tenim un sistema de planificació Round Robin amb prioritats (en cas d'empat, algorisme FIFO) amb apropiació immediata. Suposem que estem executant un procés el codi del qual cap en una pàgina (que està a memòria física). Indiqueu, **raonant breument** la vostra resposta, si en les següents situacions pot ser que el procés en RUN deixi d'estar en RUNNING, i si ho fa, en quines circumstàncies.

a) Un dels threads del procés en RUN executa una instrucció sem_wait(S)

b) El procés en RUN rep un SIGNAL

c) El procés en RUN executa una instrucció write(p[1],&c, sizeof(char)); on p[1] és el canal d'escriptura a una pipe anònima.

d) Arriba una interrupció de disc

e) Arriba una interrupció de rellotge

f) El procés en RUN executa la instrucció vector[i]+=a;

g) El procés en RUN executa un pthread_exit(...);

h) El procés en RUN executa una instrucció execlp(...)

APELLIDOS:	**DNI:**
NOMBRE:	
FILA:	**COLUMNA:**

Exercici 3: (2 punts)

Sigui un SF de tipus UNIX amb els següents i-nodes i blocs de dades.

Num_inode	2	3	4	5	6	7	8	9 (free)
Tipus	Dir	Dir	Dir	Data	Data	Data	Link	
# links	4	2	3	3	1	1	1	
Blocs de dades	12	10	14	17	13,16,11	15	18	

Bloc=10		11	12		13	14		15	16	17	18	19 (free)
.	3	Dades 1	.	2	Dades 2	.	4	Dades 3	Dades 4	Dades 5	/A/H	
..	2		..	2		..	2					
D	6		A	3		G	5					
E	5		B	4		H	7					
F	4		C	5		I	8					

a) Dibuixa l'arbre de directoris. Hi ha alguna cosa que no quadri? Creus que és un error o es pot haver arribat a aquesta situació de forma correcta? Si creus que es pot haver arribat de forma correcta, indica com; sinó, indica perquè no és correcta.

b) Suposem que l'i-node 8 és de tipus data (la resta és tot igual). Fes les estructures pel mateix arbre de directoris, però en un sistema FAT.

c) (Contesteu a la mateixa taula) Com quedarien els i-nodes i els blocs de dades després d'executar les següents instruccions (al final de les tres)

```
prompt_$ rm /B/H
prompt_$ rm /B/G
prompt_$ mv /C /A/C
```

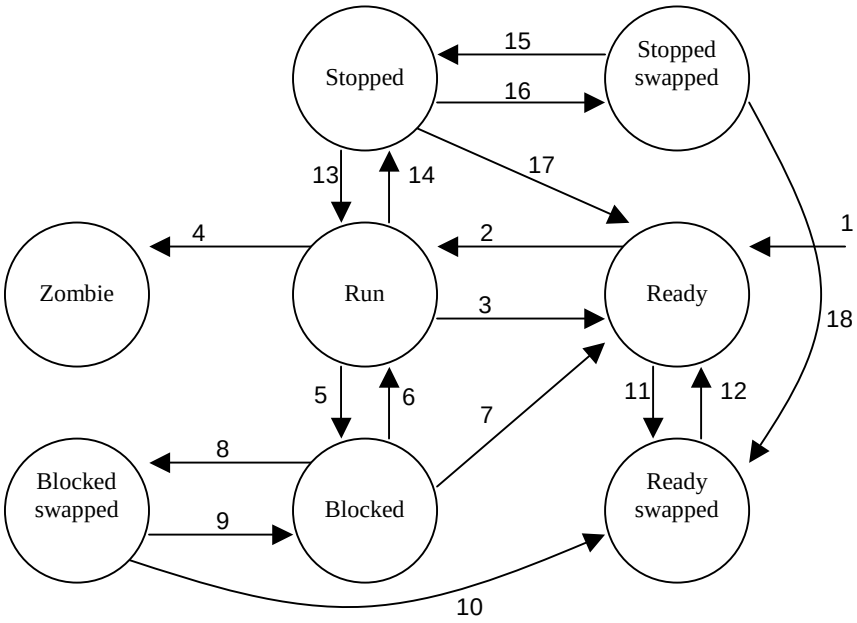
Num_inode	2	3	4	5	6	7	8	9 (free)
Tipus	Dir	Dir	Dir	Data	Data	Data	Link	
# links	4	2	3	3	1	1	1	
Blocs de dades	12	10	14	17	13,16,11	15	18	

Bloc=10		11	12		13	14		15	16	17	18	19 (free)
.	3	Dades 1	.	2	Dades 2	.	4	Dades 3	Dades 4	Dades 5	/A/H	
..	2		..	2		..	2					
D	6		A	3		G	5					
E	5		B	4		H	7					
F	4		C	5		I	8					

APELLIDOS:	**DNI:**
NOMBRE:	
FILA:	**COLUMNA:**

Exercici 4: (2 punts)

Tenim un sistema operatiu amb el següent cicle de vida:



a) Descriu qué indiquen els següents estats:

-Ready swapped:

-Stopped:

-Zombie:

b) Descriu quan es realitzen les següents transicions:

-3:

-9:

-18:

c) Modifica el graf original suposant que treiem el mecanisme de signals del nostre sistema operatiu.

d) Modifica el graf original suposant que el nostre sistema operatiu no disposa de memoria virtual ni de planificador a mig termini.

e) Modifica el graf original suposant que eliminem els gestors d'entrada/sortida del nostre sistema operatiu.

f) Quina política i algorisme de planificació está implementat al nostre sistema operatiu?

g) Amb el cicle de vida anterior, podem saber si el nostre sistema operatiu disposa de threads? Per què?

APELLIDOS:	**DNI:**
NOMBRE:	
FILA:	**COLUMNA:**

Exercici 5: (2 punts)

Tenim el següent codi que crea 2 threads que executen la mateixa funció que recorre un buffer de N elements i els processa.

```
 1: int buffer[N];
 2: int comptador = 0;
 3: void *func(void *arg) {
 4:     int index, fin=0, item;
 5:     char cadena[256];
 6:     while(!fin) {
 7:        index=comptador;
 8:        item=buffer[index];
 9:        comptador++;
10:        if ((index+1)>=N) fin=1; else {
11:            processar(item); /*Es pot executar per més d'un thread a la vegada*/
12:            sprintf(cadena, "Thread %d processa l'element %d\n", (int)arg, index);
13:            write (1, cadena, strlen(cadena)); }
14:     }
15:     pthread_exit(0);
16: }
17: int main(int argc, char **argv) {
18:     pthread_t ThA, ThB;
19:     int status;
20:     pthread_create(&ThA, NULL, func, (void*)1);
21:     pthread_create(&ThB, NULL, func, (void*)2);
22:     pthread_join(ThA, NULL);
23:     pthread_join(ThB, NULL);
24:     exit(0);
25: }
```

Volem modificar el codi, afegint semàfors, per assegurar que els threads escriguin el resultat, per la seva sortida estàndar, **per ordre d'element**. La sortida haria de ser:

```
Thread 2 processa l'element 1
Thread 2 processa l'element 2
Thread 1 processa l'element 3
Thread 2 processa l'element 4
...
```

és a dir, els elements surten ordenats però un mateix thread pot processar més d'un element consecutiu.

a) Identifica, indicant els números de línia, per la modificació proposada, les seccions crítiques de codi que han de ser protegides, raonant el motiu.

b) Reescriu el codi de la funció *func* posant els semàfors que creguis convenients, juntament amb la seva inicialització.

c) Explica breument perquè les variables *buffer* i *comptador* són globals i en canvi les variables *item* i *index* són locals.

d) Quins canvis creus que tindríem que fer al sistema operatiu per poder sincronitzar els threads d'aquesta aplicació mitjançant signals?