



## PAC1

### Presentació

Aquesta PAC planteja un seguit d'activitats amb l'objectiu que l'estudiant es familiaritzi amb la temàtica dels primers mòduls de l'assignatura.

### Competències

#### *Transversals*

- Capacitat per a la comunicació escrita en l'àmbit acadèmic i professional

#### *Específiques*

- Capacitat per a analitzar un problema en el nivell d'abstracció adequat a cada situació i aplicar les habilitats i coneixements adquirits per a abordar-lo i resoldre'l.

### Objectius

Els objectius d'aquesta PAC són que l'alumne:

- Entengui el concepte de sistema operatiu com a màquina virtual
- Conegui el cicle de vida dels processos
- Entengui la problemàtica de la gestió de memòria
- Conegui la diversitat de dispositius existents i les seves modalitats de treball.

### Enunciat

#### 1. (2.0 punts, 0.5 per apartat) [del 3/4 al 6/4]

Responen justificadament a les següents preguntes relacionades amb els mòduls 1 i 2 de l'assignatura:

- a) Que diferències hi ha entre el diagrama d'estats d'execució d'un procés en un sistema operatiu multiprogramat i un sistema operatiu monoprogramat?

**La principal diferència és que en els sistemes operatius monoprogramats no existeix l'estat de preparat/ready, ja que**



**només hi pot haver un programa en memòria i execució en cada moment. En aquest cas, el procés només deixa la CPU quan ha de fer E/S (passant a l'estat de WAIT) o finalitza.**

- b) És cert que un trap és l'únic mecanisme d'entrada al SO que pot provocar totes les transicions existents en el diagrama d'estat d'un procés? Per justificar la resposta, indicar un exemple de trap que pugui provocar cada transició d'estat i les transicions que no es pugin provocar mitjançant un trap.

**És fals. L'única transició que no pot generar és la transició de Run->Ready que es produeix quan un procés perd la CPU perquè se li ha esgotat el quantum o perquè un procés més prioritari a entrat a la cua de preparats.**

**La resta de transicions es poden produir mitjançant un trap (crida al sistema):**

- **Creació Procés (New->Ready): mitjançant la crida al sistema fork().**
- **Procés bloquejat (Run->Blocked): mitjançant qualsevol crida al sistema bloquejant (read, wait, ect).**

**Aquestes crides al sistema, també impliquen la transició de adquirir la CPU (Ready->Run), ja que sempre que la cpu queda lliure un altre procés l'ocupa.**

- **Desbloquejar procés (Blocked ->Ready): un procés bloquejat esperant la finalització d'un procés fill o la recepció d'un senyal, es pot desbloquejar quan un altre procés es finalitzi mitjançant la crida al sistema exit() o envieu una senyal amb kill() .**
- **Fi procés (Run-> End): mitjançant la crida al sistema exit().**

- c) Indiqueu si les següents instruccions es poden executar en mode no privilegiat (mode usuari) o només en mode privilegiat (mode supervisor). Justifiqueu la resposta.

- i. Generar un TRAP (interrupció programari)

**Els traps s'han de poder executar en mode usuari de manera que les aplicacions d'usuari puguin invocar a les crides al sistema. En mode privilegiat no té sentit invocar-ho, ja que el SO pot invocar directament a les seves rutines de servei.**

- ii. Deshabilitar les interrupcions



**En mode no privilegiat no s'hauria de permetre deshabilitar les interrupcions, ja que sense aquestes, els dispositius no poden avisar al SO per iniciar/completar les operacions d'E/S. Sense interrupcions, el sistema no pot funcionar, per tant és un greu risc permetre que l'usuari les pugui deshabilitar.**

**En mode privilegiat, si que s'ha de permetre deshabilitar les interrupcions. Són necessàries per evitar la generació incontrolada d'interrupcions o com a mecanisme per garantir l'exclusió mútua a nivell del SO.**

iii. Assignar un frame a l'entrada d'una taula de pàgines

**Aquesta operació només es pot executar en mode privilegiat, ja que permetria accedir a qualsevol pàgina de memòria física de qualsevol procés del sistema, això afectaria a la seguretat/protecció del sistema.**

iv. Modificar el contingut de la pila del procés.

**Aquesta operació es pot realitzar en mode no privilegiat, ja que la pila pertany a l'espai de memòria lògica del procés i per tant, l'ha de poder modificar.**

d) ¿Quines similituds i diferències hi ha entre la modalitat d'execució de l'interpret de comandes en primer pla (foreground) o de fons (background)?

**En ambdós casos l'interpret de comandes s'executa en modalitat usuari. En ambdós casos l'interpret crea un procés fill per executar les comandes demanades; en el cas background el pare segueix la seva tasca (esperar noves peticions) sense esperar el fill; en el cas foreground el pare espera a què acabi el fill. També, en ambdós casos comparteixen el terminal com a sortida estàndard dels seus programes.**



2. (2.0+4.0 punts) [del 7/4 al 14/4]

a) Contesteu justificadament a les següents preguntes:

- i. Des del punt de vista de la gestió de memòria, que implicacions té el disseny d'un SO multiprogramat respecte a un sistema operatiu monoprogramat.

**En un sistema multiprogramat cal compartir la memòria entre els diferents processos. Això implica la utilització d'un sistema de gestió de memòria (particions, segmentació o paginació) que permeti assignar un espai de memòria lògica independent per a cada procés i protegir-lo de l'accés de la resta de processos.**

- ii. Tindria sentit que un sistema operatiu per lots i monoprogramat utilitzés memòria virtual? Quins avantatges o desavantatges li proporcionaria?

**Si que tindria sentit, encara que només per augmentar la mida de memòria virtual disponible per a un procés (superar la mida de la memòria física disponible).**

**Avantatges: Suporta programes més grans i amb més requisits de memòria.**

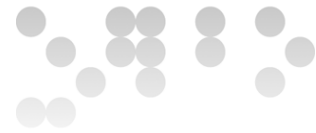
**Desavantatges: L'increment del temps d'accés i que és massa complex i costós per a un sistema operatiu monoprogramat.**

- iii. En un sistema de memòria basat en particions, si un procés realitza un accés a memòria que no pertany al procés, Com verifica el maquinari que la posició de memòria no és correcta? Què ha de passar perquè es generi una excepció d'accés a memòria?

**Comprova la direcció amb els registres límit de la partició. Si l'adreça és inferior a la direcció inicial de la partició de memòria del procés o és superior a la direcció final, llavors es genera l'error mitjançant una excepció.**

- iv. En un sistema de memòria basat en paginació, si un procés intenta realitzar un accés a una adreça invàlida, Com determina el maquinari que la posició de memòria no és vàlida? Què ha de passar perquè es generi una excepció d'accés a memòria?

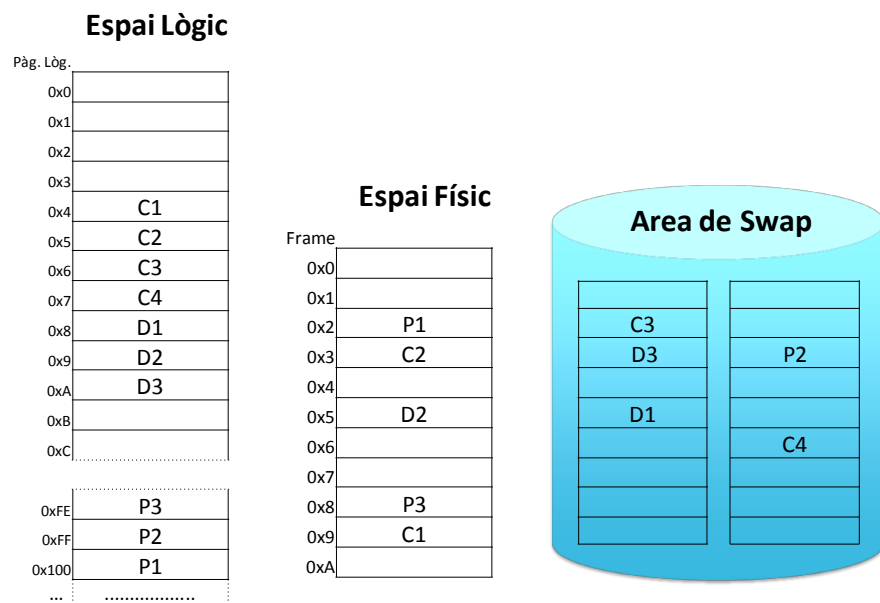
**S'accedeix a l'entrada a la taula de pàgines del procés corresponent a l'adreça lògica que està accedint el procés. Si el bit de validesa està desactivat el procés no té accés a la direcció i es genera l'excepció d'accés incorrecte a memòria.**



b) Tenim un sistema de gestió de memòria basat en paginació sota demanda, amb les següents característiques:

- Mida direcció lògica: 24 bits.
- Mida pàgina: 4 KBytes.
- Mida memòria física: 1 MByte.

Suposant que les pàgines del procés gener es carreguen en memòria física tal com indica el següent diagrama:



Assumiu que les pàgines lògiques en blanc o no mostrades són invàlides.

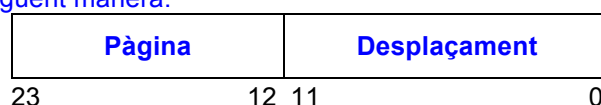
Es demana:

- Calculeu el nombre d'entrades totals a la taula de pàgines d'un procés.

Nombre entrades = Mida Memòria Lògica / Mida Pàgina = 16 MBytes / 4Kbytes =  $2^{24}/2^{12}=2^{12} = 4.096$  entrades

- Calculeu la mida dels camps de l'adreça lògica i l'adreça física.

L'adreça lògica té 24 bits,  $\log_2(16 \text{ MB})$ , que es descomponen de la següent manera:



Desplaçament pàgina =  $\log_2(\text{Mida\_pag}) = \log_2(4096) = \log_2(2^{12}) = 12$  bits



Pàgina =  $\log_2(\text{Nombre\_pags}) = \log_2(4096) = \log_2(2^{12}) = 12$  bits

La direcció física té 20 bits,  $\log_2(1\text{MB})$ , que es descomponen de la següent manera:

Frame	Desplaçament
19	12 11 0

Desplaçament pàgina =  $\log_2(\text{Tam\_pag}) = \log_2(4096) = \log_2(2^{12}) = 12$  bits

Frame =  $\log_2(\text{Nombre\_frames}) = \log_2(1\text{MB} / 4\text{Kbytes}) = 8$  bits

- iii. Calculeu el contingut de les taules de pàgines per al procés.

## Taula de Pàgines

	V	P	Frame
0x0	0	0	
0x1	0	0	
0x2	0	0	
0x3	0	0	
0x4	1	1	0x09
0x5	1	1	0x03
0x6	1	0	
0x7	1	0	
0x8	1	0	
0x9	1	1	0x05
0xA	1	0	
0xB	0	0	
0xC	0	0	
0xFE	1	1	0x08
0xFF	1	0	
0x100	1	1	0x02

- iv. A partir d'aquesta taula de pàgines, indica les adreces físiques que es correspondrien amb les següents adreces lògiques: 0x0525 i 0x9010.

**Per convertir una adreça lògica només cal indexar a la taula de pàgines amb el nombre de pàgina corresponent a l'adreça lògica i obtenir el frame de memòria virtual assignat a aquesta pàgina. A continuació es concatena el nombre de frame amb el desplaçament a la pàgina / frame i s'obté l'adreça física.**

- **Adr. Lògica 0x0525h → Excepció pàgina invalida**



**Pàgina: 0x0 Despl: 0x525h**

**Frame: TaulaPàgines[Pàg] = TaulaPàgines[0] → V = 0**

**Pàgina invalida**

- **Adr. Lògica 0x9010 → Adr. Física: 0x5010**

**Pàgina: 0x9 Despl: 0x010**

**Frame: TaulaPàgines[Pàgina] = TaulaPàgines[0x9] = 05h**

**Adr. Física: 0x05 & 0x010 = 0x5010**

- v. A partir d'aquesta taula de pàgines, indica les adreces lògiques que es correspondrien amb les següents adreces físiques: 0x2456 i 0x4000.

- **Adr. Física 0x2456 → Dir. Lògica 0x10456**

**El frame corresponent a aquesta direcció és el 0x2. A continuació es comprova a la taula de pàgines si alguna entrada conté aquest frame (l'0x10) i si aquesta entrada és vàlida i està present. Si és així, l'adreça física té correspondència amb una adreça lògica en el procés, i s'obté concatenant el nombre de pàgina de l'entrada que conté el frame (0x10) i el desplaçament de l'adreça física (0x456).**

- **Adr. Física 0x4000 → No està mapejada al espai lògic del procés**

**El frame corresponent a aquesta direcció és el 0x4. A continuació es comprova a la taula de pàgines si alguna entrada conté aquest frame. Com que no existeix aquesta entrada, vol dir que el procés no té assignada aquesta adreça física.**

### 3. (2.0 punts, 1.0 per apartat) [del 15/4 al 17/4]

Respondre justificadament a les següents preguntes relacionades amb el mòdul 4 de l'assignatura:

- a) Si executem la següent comanda "pgm1 2> /mnt/pen/file.txt", estant file.txt en un dispositiu extraïble. En quin descriptor està escrivint el programa pgm1? Si retirem el pendrive abans que es completi l'operació d'E/S, com s'assabenta el programa d'aquesta situació?

**En aquest cas, la sortida d'error del procés pgm1 està redirigida cap al fitxer file.txt. Per tant el programa pgm1 està escrivint a la stderr, descriptor 2.**

**En el cas que es produeixi qualsevol error i el programa no pugui escriure en el dispositiu, se notifica al mateix mitjançant el valor de retorn de la crida a sistema write.**



b) Indiqueu quines característiques d'accés (compartit/exclusiu, seqüencial/directe/indexat, síncron/ asíncron i cuinat / cru) tenen els següents dispositius:

i. pipe

**Accés compartit (més d'un procés pot llegir / escriure en una pipe). Accés seqüencial. Accés síncron i asíncron en funció del mode que esculli el programador. Mode escriptura en cru.**

ii. stdout

**Accés compartit (més d'un procés pot escriure simultàniament en la stdout). Accés seqüencial. Accés síncron i asíncron en funció del mode que esculli el programador. Mode escriptura cuinat (suporta comandes per afegir salts de línia, modificar el color, de posicionat, etc.), depenent del tipus de dispositiu físic al com pugui estar redirigit la stdout.**

iii. Un fitxer

**Accés compartit (més d'un procés pot llegir / escriure en un fitxer). Accés seqüencial, directe i indexat. Accés síncron i asíncron en funció de la manera d'obertura. Mode escriptura en cru.**

## Recursos

### Bàsics

- Document “Intèrpret de comandes Unix”, disponible a l'aula virtual
- Mòduls 1, 2, 3 i 4

### Complementaris

- La bibliografia de l'assignatura

## Criteris de valoració

Es valorarà la justificació de les respostes presentades. S'agrairan les respostes breus i concises. El pes de cada pregunta està indicat a l'enunciat.

En la correcció es tindran en compte els següents aspectes:

- Les respostes hauran d'estar articulades a partir dels conceptes estudiats en teoria i en la guies de l'assignatura.
- S'agrairà la claredat i la capacitat de síntesis en les respostes.





- Es valorarà essencialment la correcta justificació de les respostes, recolzada pels fonaments teòrics.

## Format i data de lliurament

La solució es lliurarà en un fitxer text (format .txt o .pdf)

El nom del fitxer tindrà el format següent:

"Cognom1Cognom2PAC2.txt". Els cognoms s'escriuran sense accents.

Per exemple, l'estudiant Marta Vallès i Marfany utilitzarà el nom de

**fitxer següent: VallesMarfanyPAC2.txt**

La data límit per al lliurament de la activitat és el **dimecres 17 d'abril de 2019**.