

DS Systèmes d'exploitation

Durée 1h30, sans documents. Le barème est indicatif.

A. Systèmes RAID – 8 pts.

On dispose de trois disques de 500 Go à raccorder à un système multi-tâches.

Question A1. À l'aide d'un schéma, montrez comment sont répartis les blocs 0, 1, 2, 3, 4, 5... si on utilise les 3 disques

- en RAID0 (“striping”) à 3 disques
- en RAID1 (miroir) à 3 disques

Indiquez la *capacité* de chaque solution.

Question A2. Pour comparer la *fiabilité* de ces deux solutions, vous calculerez la *probabilité d'une panne* du système RAID0, sachant que chaque disque a 1 chance sur 10 de tomber en panne dans l'année (ou plutôt 9 chances sur 10 de bien fonctionner).

Question A3. Même question pour le RAID1 à trois disques.

Question A4. Supposons que la lecture ou l'écriture d'un bloc sur un disque prenne en moyenne 10 ms. Sur un système RAID0 à 3 disques combien pourra-t-on faire, au maximum, de lectures par seconde? D'écritures par seconde? Dans chaque cas, montrez par un exemple quelles opérations peuvent se faire simultanément.

Question A5. Même question pour un RAID1 à 3 disques.

Question A6. Montrez comment faire du RAID 0+1 avec 4 disques (RAID1 au dessus de RAID0).

Question A7. On peut aussi utiliser les 4 disques en RAID 1+0, avec des performances semblables. Comparez la fiabilité des deux solutions RAID 0+1 et RAID 1+0. Vous détaillerez le calcul de probabilités de panne. Conclusion?

B. Adressage – 7 pts.

Un de vos clients fait appel à vous en tant qu'expert informatique sur architectures exotiques. Un des processus de son serveur n'arrête pas de planter. Le serveur en question est un ordinateur à **données 8 bits** et **adresses physiques 16 bits** d'un modèle inconnu qui supporte segmentation et pagination. Le client parvient à faire un *dump mémoire*¹ du processus fautif et des tables du système d'exploitation, et vous l'envoie.

En tant qu'expert, vous localisez en premier la table des segments du processus, et la table des pages. La notation 0x précède un nombre écrit en base 16 (hexadécimal), chaque chiffre hexadécimal (0, 1, 2, ... A, B, C, D, E, F) est codé sur 4 bits.

	Début	Fin
0x0	0x0000	0x8000
0x1	0x1000	0x2000
0x2	0x2000	0x4000
0x3	0xF000	0xFFFF
0x4	0x0000	0xFFFF
0x5	0x0100	0x0200
0x6	0x0000	0x0000
0x7	0x0000	0x0000

TAB. 1 – Table des segments

	Adresse Ram ou SWAP	Bit P
0x0	swap :0x4567	0
0x1	0x0000	1
0x2	0x2000	1
0x3	NOT PRESENT	0
0x4	0x3000	1
0x5	0x5000	1
0x6	0x6000	1
0x7	0xA000	1
0x8	0xC000	1
0x9	0x0000	1
0xA	swap :0x6745	0
0xB	0xB000	1
0xC	0xB000	1
0xD	0x1000	1
0xE	0x4000	1
0xF	NOT PRESENT	0

TAB. 2 – Table des pages

Question B1. Sachant qu'une adresse virtuelle est codée sur **19 bits**, qu'elle est de la forme SEGMENT | PAGE | DÉPLACEMENT, et d'après la taille des tables de segments et de pages, donnez le format détaillé d'une adresse virtuelle.

Question B2. Détaillez le raisonnement qui vous permet de déterminer la taille d'une page mémoire (et indiquez-la).

¹affichage d'une partie de la mémoire

Question B3. À quoi sert le bit P de la table des pages ?

Question B4. Vous avez repéré quatre instructions suspectes dans le code du programme fautif. Pour chacune d'elles, expliquez brièvement pourquoi elle peut (ou ne peut pas) être responsable du crash du programme (Rappel : la segmentation a lieu **avant** la pagination).

1. lecture d'un octet à l'adresse virtuelle 0x1E256
2. lecture d'un octet à l'adresse virtuelle 0x00345
3. lecture d'un octet à l'adresse virtuelle 0x21823
4. lecture d'un octet à l'adresse virtuelle 0x4EE34

C. Gestion des entrées-sorties – 5 pts.

Soient 3 processus P1, P2 et P3 arrivés dans cet ordre à peu près au même instant sur un système multitâche. Ils se comportent tous les trois de façon cyclique. P1 fait du calcul pendant un temps t , puis lit le bloc numéro 100, puis calcule, lit le bloc 101, calcule, lit le bloc 102, etc. Il en est de même pour P2 et P3, qui font leurs lectures à partir des blocs 500 et 300 respectivement.

Dans tout l'exercice, l'ordonnancement des processus est fait par l'algorithme du *tour-niquet*. Le temps d'exécution d'une opération d'entrée-sortie à un endroit donné dépend de la position précédente. La formule

$$\text{durée} = 5 + (\text{déplacement}/10)\text{ms}$$

indique le temps nécessaire en fonction du déplacement, exprimé en nombre de blocs. La division est arrondie. Par exemple la lecture du bloc 123 après celle du bloc 12 prend $5 + (123 - 12)/10 = 16\text{ms}$.

Question C1. Calculez le temps nécessaire pour chacune des transitions

$$\begin{aligned}t_0 &: 0 \rightarrow 100 \\t_1 &: 100 \rightarrow 500 \\t_2 &: 500 \rightarrow 300 \\t_3 &: 101 \rightarrow 501 \\t'_1 &: 501 \rightarrow 301\end{aligned}$$

Question C2. Étudiez les 100 premières ms du comportement de l'ordonnancement "FI-FO" pour les entrées-sorties sur disque, quand le temps de calcul est de $t=3$ ms. Initialement la tête de lecture est en position 0. Vous donnerez un graphique à l'échelle 1 cm / 10 ms, qui présentera clairement les états des processus, et les requêtes en attente.

Question C3. même question avec l'ordonnancement "plus court déplacement". Que se passe-t'il ensuite ?