

	<p align="center">Examen Session Normale 21/12/2023</p>	
M. Allouch, A. Bertails, T. Guidini, J.A. Lorenzo, M. Zneika	Système d'exploitation	
ING1 Informatique-Mathématiques Appliquées	Année 2023–2024	

Modalités

- Durée : 2 heures.
- Vous devez rédiger votre copie à l'aide d'un stylo à encre exclusivement.
- Toutes vos affaires (sacs, vestes, trousse, etc.) doivent être placées à l'avant de la salle.
- Aucun document n'est autorisé.
- La calculatrice est autorisée.
- Le QCM peut contenir une ou plusieurs réponses valides.
- Aucune question ne peut être posée aux enseignants, posez des hypothèses en cas de doute.
- Aucune sortie n'est autorisée avant une durée incompressible d'une heure.
- Aucun déplacement n'est autorisé.

QCM à répondre sur votre copie et non pas sur le sujet : (6 points)

a) Un système d'exploitation...

- ☐ doit toujours fournir une réponse en temps réel
- ☐ s'occupe de la résolution des conflits entre processus
- ☐ a le contrôle sur la mémoire caché
- ☐ Aucune des réponses n'est valide

b) L'espace d'échange (swap)...

- ☐ est utilisé lorsque la mémoire RAM arrive à saturation
- ☐ peut contenir des processus zombie, en attente et prêts
- ☐ se trouve généralement sur le stockage secondaire
- ☐ Aucune des réponses n'est valide

c) En cas de défaut de page...

- ☐ la victime sera toujours écrite sur le disque

- ☐ une nouvelle page virtuelle est chargée en mémoire physique
- ☐ le *dirty bit* est mis à jour
- ☐ Aucune des réponses n'est valide

d) L'allocation indexée...

- ☐ ne permet que l'accès séquentiel d'une donnée
- ☐ génère une fragmentation du disque importante
- ☐ peut difficilement grandir
- ☐ Aucune des réponses n'est valide

e) Nous utilisons la ligne de commande pour démarrer un programme à interface graphique. Pendant que ce programme tourne, la ligne de commande reste bloquée (nous ne pouvons pas introduire plus de commandes). En termes de processus père et fils, quelle est la raison pour laquelle la ligne de commande est bloquée jusqu'à la terminaison du programme graphique ?

- ☐ Le processus père bloque le processus fils jusqu'à la terminaison.
- ☐ Le processus fils bloque le processus père jusqu'à la terminaison.
- ☐ La ligne de commande est indépendante du programme graphique.
- ☐ Le processus père et le processus fils sont complètement isolés.

f) Quelle est la différence entre Pile (Stack) et Tas (Heap) ?

- ☐ La pile est utilisée pour stocker des données statiques, tandis que le tas est utilisé pour des données dynamiques.
- ☐ La pile est gérée par le programme, tandis que le tas est géré par le système d'exploitation.
- ☐ La pile est de taille fixe, tandis que le tas est de taille dynamique.
- ☐ La pile stocke les variables locales et les appels de fonctions, tandis que le tas est utilisé pour la gestion dynamique de la mémoire.

g) Combien de partitions peut-il avoir au maximum dans l'organisation logique d'un disque MBR ?

- ☐ Quatre partitions primaires
- ☐ Trois partitions primaires
- ☐ Trois partitions primaires et une étendue
- ☐ Aucune des réponses n'est valide

h) Qu'est-ce que la mémoire virtuelle ?

- ☐ La mémoire RAM physique installée sur un ordinateur
- ☐ Un type de stockage externe
- ☐ Une extension de la mémoire RAM utilisant le disque dur
- ☐ Une mémoire utilisée uniquement par les machines virtuelles

Exercice 1 : Programmation de processus (2 points)

Concernant le programme ci-dessous, quels sont les messages affichés par le programme et dans quel ordre ?

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <sys/types.h>
4  #include <sys/wait.h>
5  #include <unistd.h>
6
7  void child(int number) {
8      printf("Je suis le processus %d\n", number);
9      exit(0);
10 }
11
12 int main() {
13     pid_t pid1, pid2;
14     int status;
15
16     if ((pid1 = fork()) == 0) {
17         child(1);
18     }
19
20     if ((pid2 = fork()) == 0) {
21         child(2);
22     }
23
24     waitpid(pid1, &status, 0);
25     waitpid(pid2, &status, 0);
26
27     printf("Tous les processus sont terminés\n");
28
29     return 0;
30 }
```

Exercice 2 : Ordonnancement (4 points)

Considérons cinq processus A, B, C, D et E, dont la durée d'exécution et le temps d'arrivée respectifs sont donnés dans le tableau ci-dessous.

Processus	Durée	Arrivée
A	3	0
B	8	6
C	4	9
D	1	11
E	1	4

- Faire un schéma qui illustre l'exécution de ces différents processus pour chaque technique d'ordonnancement : **NON Préemptif** avec FCFS (First Come First Served), avec SJF (Shortest Job First), et avec RR (Round Robin, quantum = 2 unités de temps) ; et **Préemptif** avec SRT (Shortest Remaining Time).
- Calculer le temps de séjour de chaque processus, le temps moyen de séjour, le temps d'attente et le temps moyen d'attente, pour chaque technique d'ordonnancement.
- L'efficacité moyenne permet au système d'exploitation de choisir l'ordonnanceur à utiliser pour la liste de processus. Calculer l'efficacité pour chaque processus (**temps de séjour du processus / durée d'exécution du processus**) ainsi que l'efficacité moyenne par ordonnanceur. Quel ordonnanceur sera utilisé par le système d'exploitation ?

Exercice 3 : Mémoire virtuelle (3 points)

Considérez un système avec une architecture de mémoire virtuelle et physique. La mémoire virtuelle utilise une taille de page de 4Mo, et la mémoire physique est composée de cadres de page. Les adresses virtuelles se font sur 32 bits et les adresses physiques (réels) sur 28 bits.

- Quelle est la taille de page en octets ?
- Combien de bits sont utilisés pour l'offset dans l'adresse virtuelle et dans l'adresse physique ?
- Combien de pages peuvent être adressées dans la mémoire virtuelle ?
- Combien de bits sont nécessaires pour adresser chaque cadre de page dans la mémoire physique ?
- Combien de cadres de page peuvent être adressés dans la mémoire physique ?

Exercice 4 : Le système de fichiers (3 points)

L'indexation des blocs mémoire dans un système de fichiers EXT4 est définie comme suit :

- Les 12 premiers champs correspondent à des blocs directs ;
 - Le 13ème champ correspond à de blocs indirects simple ;
 - Le 14ème champ correspond à de blocs indirects double ;
 - Le 15ème champ correspond à de blocs indirects triple.
-
- a) Comment peut-on calculer la taille maximale d'un fichier dans ce système de fichier EXT4 ?
 - b) Sachant que nous avons des blocs de taille 3 Ko qui utilisent des numéros de blocs sur 24 bits, quelle est la taille maximale d'un fichier ?

Exercice 5 : Docker (2 points)

Sachant que je n'ai pas encore utilisé docker sur ma machine, je vais exécuter successivement les commandes :

```
root@MachineCyTech> docker run hello-world  
root@MachineCyTech> docker run hello-world
```

- a) Que se passe-t-il lors de l'exécution de la première commande ?
- b) Que se passe-t-il lors de l'exécution de la deuxième commande ?
- c) Quelles commandes utiliser pour lister les images et les containers ?
- d) Combien d'images et combien de containers sont présents sur la machine ?

CORRIGÉ

Questions QCM (0.75 par question)

- a) Un système d'exploitation...
 - ☐ s'occupe de la résolution des conflits entre processus

- b) L'espace d'échange (swap)...
 - ☐ est utilisé lorsque la mémoire RAM arrive à saturation
 - ☐ se trouve généralement sur le stockage secondaire

- c) En cas de défaut de page...
 - ☐ une nouvelle page virtuelle est chargée en mémoire physique

- d) L'allocation indexée...
 - ☐ Aucune des réponses n'est valide

- e) Nous utilisons la ligne de commande pour démarrer ...
 - ☐ Le processus fils bloque le processus père jusqu'à la terminaison.

- f) Expliquer la différence entre les régions de la mémoire Pile (Stack) et Tas (Heap)
 - ☐ La pile stocke les variables locales et les appels de fonctions, tandis que le tas est utilisé pour la gestion dynamique de la mémoire.

- g) Combien de partitions peut-il avoir au maximum dans l'organisation logique d'un disque MBR ?
 - ☐ Quatre partitions primaires
 - ☐ Trois partitions primaires et une étendue

- h) Qu'est-ce que la mémoire virtuelle ?
- ☐ Une extension de la mémoire RAM utilisant le disque dur

Exercice 1 : Programmation de processus (2 points)

Le résultat possible:

Message du processus 1 (premier processus enfant) : Je suis le processus 1

Message du processus 2 (deuxième processus enfant) : Je suis le processus 2

Message "Tous les processus sont terminés" (du processus parent).

Le message du processus parent s'affiche toujours après la terminaison des deux processus enfants.

Les messages des processus enfants peuvent apparaître dans un ordre différent.

Exercice 2 : Ordonnancement (4 points)

- a) Faire un schéma qui illustre l'exécution de ces différents processus pour chaque technique d'ordonnancement : **NON Préemptif** avec FCFS (First Come First Served), avec SJF (Shortest Job First), et avec RR (Round Robin, quantum = 2 unités de temps) ; et **Préemptif** avec SRT (Shortest Remaining Time) (2 points / 0.5 pour chaque technique).
- b) Calculer le temps de séjour de chaque processus, le temps moyen de séjour, le temps d'attente et le temps moyen d'attente, pour chaque technique d'ordonnancement (1 point / 0.25 pour chaque technique).
- c) L'efficacité moyenne permet au système d'exploitation de choisir l'ordonnanceur à utiliser pour la liste de processus. Calculer l'efficacité pour chaque processus (**temps de séjour du processus / durée d'exécution du processus**) ainsi que l'efficacité moyenne par ordonnanceur. Quel ordonnanceur sera utilisé par le système d'exploitation (1 point / 0.2 pour chaque technique et 0.2 pour la bonne réponse concernant l'ordonnanceur) ?

Le SRT sera utilisé par le système d'exploitation.

FCFS

Processus	Durée	Arrivée ->
1 A	3	0
2 B	8	6
3 C	4	9
4 D	1	11
5 E	1	4
Moyenne	3,4	Temps ->

Fin	Séjour	Attente	Efficacité
3	3	0	100,00 %
14	8	0	100,00 %
18	9	5	44,44 %
19	8	7	12,50 %
5	1	0	100,00 %
Moyenne	5,8	2,4	71,39 %

SJF

Processus	Durée	Arrivée ->
1 A	3	0
2 B	8	6
3 C	4	9
4 D	1	11
5 E	1	4
Moyenne	3,4	Temps ->

Fin	Séjour	Attente	Efficacité
3	3	0	100,00 %
14	8	0	100,00 %
19	10	6	40,00 %
15	4	3	25,00 %
5	1	0	100,00 %
Moyenne	5,2	1,8	73,00 %

RR avec un Quantum de 2

Processus	Durée	Arrivée ->
1 A	3	0
2 B	8	6
3 C	4	9
4 D	1	11
5 E	1	4
Moyenne	3,4	Temps ->

Fin	Séjour	Attente	Efficacité
3	3	0	100,00 %
19	13	5	61,54 %
16	7	3	57,14 %
12	1	0	100,00 %
5	1	0	100,00 %
Moyenne	5	1,6	83,74 %

SRT

Processus	Durée	Arrivée ->
1 A	3	0
2 B	8	6
3 C	4	9
4 D	1	11
5 E	1	4
Moyenne	3,4	Temps ->

Fin	Séjour	Attente	Efficacité
3	3	0	100,00 %
19	13	5	61,54 %
14	5	1	80,00 %
12	1	0	100,00 %
5	1	0	100,00 %
Moyenne	4,6	1,2	88,31 %

Exercice 3 : Mémoire (3 points) 0.6 pour chaque question

- a) $4 \text{ MB} = 4 * 1024 * 1024 = 4194304 \text{ bytes}$
- b) $4\text{Mo} = 2^2 * 2^{20} = 2^{22}$, alors 22 bits pour OFFSET
- c) MV, $32 - 22 = 10$, alors 2^{10} ou **1024 (1K)** pages en MV
- d) MP, $28 - 22 = 6$, alors 6 bits
- e) $2^6 = 64$ cadres de page

Exercice 4 : Le système de fichiers (3 points)

L'indexation des blocs mémoire dans un système de fichiers EXT4 est définie comme suit :

- Les 12 premiers champs correspondent à des blocs directs ;
- Le 13ème champ correspond à de blocs indirects simple ;
- Le 14ème champ correspond à de blocs indirects double ;
- Le 15ème champ correspond à de blocs indirects triple.

- a) Comment peut-on calculer la taille maximale d'un fichier dans ce système de fichier EXT4 ? (1.0 point)

La formule pour calculer la taille maximale d'un fichier est donnée par :

Taille maximale du fichier = (Nombre de blocs directs + Nombre de blocs indirects¹ + Nombre de blocs indirects² + Nombre de blocs indirects³) * Taille du bloc

- b) Sachant que nous avons des blocs de taille 3 Ko qui utilisent des numéros de blocs sur 24 bits, quelle est la taille maximale d'un fichier ? **(2.0 points)**

Dans le cas d'une taille de bloc de 3 Ko = 3*1024 o, et en considérant que chaque numéro de bloc est sur 24 bits = 3 o, nous pouvons stocker :

3 ko / 3 o = 3*1024 o / 3 o = 1024 numéros de blocs dans un bloc. **(1.0 point)**

Taille maximale du fichier =(10 + 1024 + 1024² + 1024³) * 3 Ko

= 1074791434 * 3 Ko ~ 3 To ou 3 Tb **(1.0 point)**

Exercice 5 : Docker (2 points)

Sachant que je n'ai pas encore utilisé docker sur ma machine, je vais exécuter successivement les commandes :

```
root@MachineCyTech> docker run hello-world
root@MachineCyTech> docker run hello-world
```

- a) Que se passe-t-il lors de l'exécution de la première commande **(0,5 point)** ?

Téléchargement de l'image hello-world
Création d'un container
Exécution du hello-world

- b) Que se passe-t-il lors de l'exécution de la deuxième commande **(0,5 point)** ?
Pas besoin de télécharger à nouveau l'image qui est déjà présente

Création d'un container
Exécution du hello-world

- c) Quelles commandes utiliser pour lister les images, les containers **(0,5 point)** ?

pour les images : docker images
pour les container : docker ps -a

- d) Combien d'images, combien de containers sont présentes sur la machine **(0,5 point)** ?

1 image, 2 containers

