

Préparation à l'examen de *Systèmes d'exploitation*

Wéry Benoît

9 novembre 2017

Chapitre 1

Définitions de concepts

1. *Définissez ce qu'est le système d'exploitation d'un système informatique. En particulier, identifiez ses buts et les abstractions du système informatique qu'il permet d'opérer. Présentez également la structure d'un système informatique et où il s'y situe.*
2. *Quelles sont les principales étapes qui sont franchies lors du démarrage d'un système informatique ? Citez-les et identifiez la fonction de chacune des étapes, c'est-à-dire le rôle des actions réalisées pour le fonctionnement du système informatique.*
3. *Citez et expliquez des exemples de services offerts par un système d'exploitation. Comment peut-on les classer selon le type destinataire du service ? Citez et expliquez également les trois aspects sur lesquels il joue le rôle de coordinateur.*
4. *Qu'est-ce-que la multiprogrammation ? Quelles conséquences cette tendance a-t-elle eue sur le développement des systèmes d'exploitation ?*
5. *Citez et expliquez les deux modes d'exécution possible d'un programme. Qu'est-ce-qu'un appel système et comment se déroule l'exécution d'un tel appel ?*
6. *Définissez le concept de processus et détaillez les différentes opérations que l'on peut réaliser depuis sa création jusque sa terminaison. Citez et expliquez quels sont les différents états possibles d'un processus.*
7. *Comment et où le système d'exploitation maintient-il de l'information par rapport aux processus ? Quel genre d'informations garde-t-il et pourquoi ?*
8. *Quelles sont les différences entre parallélisme et concurrence ? Comment les obtient-on dans un*

système informatique donné ? Le hardware disponible est-il un facteur déterminant dans le choix entre parallélisme et concurrence lors du développement d'un programme ?

9. *Citez et caractérisez les deux types de threads en fonction du niveau où le support des threads est fourni. Citez, expliquez et comparez les différents mappings que l'on peut réaliser entre ces deux types de thread.*
10. *Définissez ce qu'est et ce que fait l'ordonnanceur de processus, et le dispatcher. Quand ces derniers sont-ils exécutés ? Qu'est-ce que la préemption et quels sont ses avantages et inconvénients ?*
11. *Définissez ce qu'est une section critique et dans quelle situation l'existence d'une telle section peut poser problème (illustrez avec un exemple). Citez et expliquez une solution que l'on peut apporter pour protéger de telles sections en synchronisant des processus.*
12. *Définissez ce qu'est un deadlock et dans quelles situations il peut se produire (illustrez avec un exemple). Comment peut-on détecter, prévenir et se remettre d'un deadlock ?*
13. *Citez et expliquez les différents moyens que l'on peut mettre en oeuvre pour faire communiquer entre eux des processus coopératifs ? Quels sont les avantages et inconvénients de ces deux moyens de communication ?*
14. *Expliquez ce qu'est le principe d'adressage ? Comment l'unité mémoire intervient-elle ? Définissez et expliquez le principe de liaison des adresses.*
15. *En quoi consiste le swapping ? Citez et expliquez les différentes formes de swapping qu'il existe, et comparez-les.*
16. *Citez, expliquez et comparez les différentes stratégies d'allocation de la mémoire que l'on peut mettre en place. Quels sont les risques de fragmentation liés à ces stratégies ?*
17. *Expliquez le principe de la segmentation. En quoi est-il plus proche de la pensée du programmeur ? Comment les adresses sont-elles construites lorsqu'on utilise la segmentation ?*
18. *Expliquez le principe de la pagination. Quels sont ses avantages par rapport à la segmentation ? Comment les adresses sont-elles construites lorsqu'on utilise la pagination ? Comment peut-on*

améliorer les performances de la pagination à l'aide d'une mémoire spécialisée ?

19. *Citez et expliquez les différentes structures que l'on peut utiliser pour stocker la table des pages. Quels sont les avantages et inconvénients de ces structures ? Dans quelles situations sont-elles plus adaptées ?*
20. *Définissez et expliquez les différences et les liens entre mémoire physique et mémoire virtuelle. Comment le système d'exploitation fait-il le lien entre ces deux mémoires, pour les différents processus ?*
21. *Définissez et expliquez le concept de défaut de page. Quand un défaut de page se produit-il ? En quoi altèrent-ils les performances du système ?*
22. *Définissez, expliquez et comparez les différentes stratégies que l'on peut mettre en place pour allouer les cadres.*
23. *Définissez la notion d'écroulement et expliquez en quoi elle rend un système instable. Comment peut-on détecter et se protéger d'un écroulement ?*
24. *Définissez la notion de fichier et expliquez comment le système d'exploitation les gère. Citez et expliquez des exemples d'opérations qu'il est possible de réaliser sur des fichiers.*
25. *Expliquez comment un disque peut être structuré en partition/volume/systèmes de fichiers. Citez, expliquez et comparez différentes manières de concevoir une structure en répertoires.*
26. *Définissez ce qu'est un système de fichiers et les informations que le système d'exploitation retient pour l'organiser. En quoi consiste le montage d'un système de fichiers et qu'est-ce que le système de fichiers virtuel ?*
27. *Citez, expliquez et comparez les différentes méthodes d'allocation utilisées pour stocker les fichiers sur le disque.*
28. *Citez et expliquez les deux types de formatage d'un disque qu'il est possible de réaliser. Quels sont les buts précis de chacun de ces deux formatages ?*

29. *Citez, expliquez et comparez les différentes structures RAID qu'il est possible de mettre en oeuvre. Dans quel cas concret utiliseriez-vous l'une ou l'autre structure ?*
30. *Définissez ce qu'est un driver de périphérique. Où ce dernier intervient-il et avec quels autres composants interagit-il afin de permettre à un utilisateur d'exploiter un périphérique.*
31. *Citez et expliquez la notion de machine virtuelle et quelles sont les deux types de machines virtuelles qu'il est possible de mettre en oeuvre.*
32. *Quelles sont les différences entre moniteur de machines virtuelles et système d'exploitation ? En particulier, quelles sont les opérations qu'ils doivent tous les deux faire et celles qui les distinguent.*
33. *Quelles sont les similitudes et différences entre les deux types d'hyperviseur. Quels sont les avantages et inconvénients de ces deux types d'hyperviseur ?*
34. *Quelles sont les différences entre le kernel Linux, un système Linux et une distribution Linux. Mettez votre comparaison en parallèle avec les trois composants principaux de Linux qui sont en accord avec Unix.*
35. *Définissez et expliquez ce que sont les modules kernel. En particulier, expliquez à quoi ils servent, en quoi ils permettent d'obtenir un kernel minimal et en quoi ils facilitent le développement de drivers.*
36. *La création d'un nouveau processus en Linux passe par deux appels systèmes. Citez-les et expliquez leur rôle. Quels avantages apporte un tel choix de design en comparaison à Windows où un nouveau processus est directement créé avec un unique appel à l'appel système `CreateProcess`.*
37. *Expliquez le modèle CIA+ en détaillant chacun des objectifs visés.*
38. *Donnez et expliquez trois principes de conception pour le développement de mécanismes de protection.*
39. *Expliquez comment construire un certificat de clé publique, et comment on peut vérifier l'au-*

thenticité de la clé.

40. *Expliquez ce qu'apporte l'utilisation du « sel » dans le stockage de mots de passe.*
41. *Expliquez le contrôle d'accès aux fichiers Unix classique, ainsi que son extension en « Access List ».*

Chapitre 2

Raisonnement

1. *Citez, expliquez et comparez différentes approches possibles pour structurer le code d'un système d'exploitation. Quels sont les avantages et inconvénients de ces différentes approches en fonction du système informatique à opérer ? Argumentez.*
2. *Définissez ce que sont les threads et comparez-les par rapport aux processus. Que permettent-ils de plus, par rapport à un système d'exploitation ne proposant que des processus ? Confondre les threads et les processus en un seul type d'entité comme le fait Linux est-il intéressant ? Argumentez.*
3. *Citez, expliquez et comparez différentes stratégies d'ordonnancement que l'on peut mettre en oeuvre (FCFS, SJF, priorité, RR, files multi-niveaux). Comment peut-on les comparer ? Y a-t-il un meilleur algorithme dans l'absolu ? Argumentez.*
4. *L'acquisition de ressources par des processus peut conduire le système dans des états de deadlock. Il existe plusieurs stratégies permettant d'éviter ou de se remettre d'un deadlock. Quelles sont-elles et en quoi elles sont plus ou moins adaptées selon le type de système informatique ? Argumentez.*
5. *L'utilisation de threads en lieu et place de processus facilite-t-il la communication entre ces entités ? Est-on limités aux mêmes mécanismes de communication ? Où y gagne-t-on et que perd-on ? Argumentez.*
6. *Citez, expliquez et comparez différentes stratégies de remplacement des pages que l'on peut mettre en oeuvre (FIFO, optimal, LRU, approximation de LRU, LFU, MFU). Comment peut-on les comparer ? Y a-t-il un meilleur algorithme dans l'absolu ? Argumentez.*
7. *Un système d'exploitation peut adopter plusieurs attitudes différentes suite à l'apparition d'un*

deadlock. Citez-les et expliquez en quoi elles sont plus adaptées en fonction du système d'exploitation à opérer. Argumentez.

8. *Citez, expliquez et comparez différentes stratégies d'ordonnancement de disque l'on peut mettre en oeuvre (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK). Comment peut-on les comparer ? Y a-t-il un meilleur algorithme dans l'absolu ? Argumentez.*
9. *Une virtualisation est réussie si le système d'exploitation est incapable de savoir s'il tourne sur une machine physique ou sur une virtuelle. Expliquez comment un système d'exploitation pourrait se rendre compte qu'il est trompé et ce qu'on peut mettre en oeuvre pour le tromper.*
10. *Dans plusieurs situations (ordonnancement de processus, remplacement de pages, ordonnancement d'opérations disque), plusieurs algorithmes sont possibles. Y a-t-il à chaque fois un algorithme qui est le meilleur dans tous les cas ? Comment peut-on comparer ces algorithmes ? Sont-ils bon ou mauvais selon le système informatique à opérer ? Comment choisir du mieux possible l'algorithme à utiliser étant donné un système d'exploitation ? Argumentez.*
11. *Vous devez installer un serveur web sécurisé, sur base d'un OS Linux. Comment procédez-vous ? Détaillez les grandes lignes de votre stratégie de hardening.*