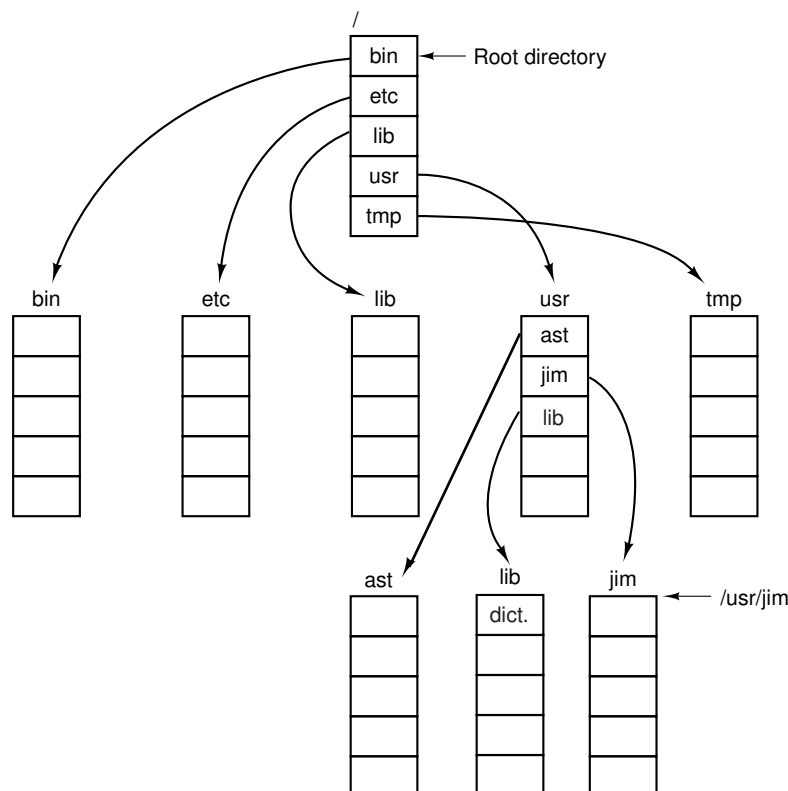


# EX04 – Systèmes de gestion des fichiers

## Systèmes d'exploitation / Classes T-2adfg

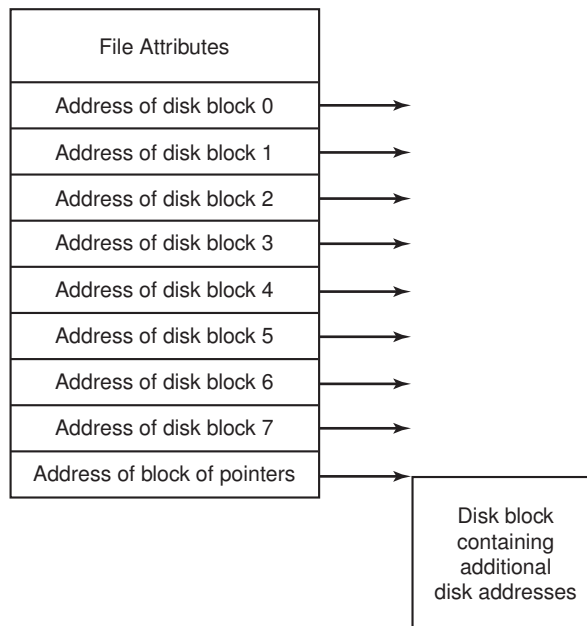
1. L'appel système «open» est-il absolument essentiel sous UNIX ? Quelles seraient les conséquences si'il n'existait pas ?
2. Les systèmes qui supportent les fichiers séquentiels ont toujours une opération pour rembobiner les fichiers. Les systèmes qui supportent des fichiers à accès direct ou aléatoire en ont-ils aussi besoin ?
3. Certains systèmes d'exploitation proposent un appel système «rename» pour donner un nouveau nom au fichier. Existe-t-il une différence entre cette opération et celle qui consiste à faire une copie du fichier dans un nouveau fichier avec un nouveau nom et à ensuite effacer le premier fichier ?
4. Un système d'exploitation élémentaire dispose uniquement d'un seul répertoire, mais lui permet d'avoir beaucoup de fichiers avec des noms longs. Peut-il simuler une sorte de système de fichiers hiérarchique ? Comment ?
5. Considérons l'arborescence de la figure ci-dessous. Si le répertoire de travail est `/usr/jim`, quel est le chemin d'accès absolu pour les fichiers dont le chemin d'accès relatif est `../ast/x` ?



6. Pour utiliser une allocation contiguë de fichiers sans pâtir des trous, il est possible de compacter le disque chaque fois qu'un fichier est effacé. Comme tous les fichiers sont contigus, la copie d'un fichier implique un déplacement du bras et un délai de rotation pour lire le fichier, puis un transfert à pleine vitesse. De même pour la réécriture du fichier. En supposant que le temps de déplacement est de 5 ms, que le délai de rotation est de 4 ms, que le taux de transfert est de 8 MB/s et que la

taille moyenne d'un fichier est de 8 KB, combien de temps faut-il pour lire un fichier depuis le disque vers la mémoire et le réécrire dans un autre emplacement du disque ? Avec ces mêmes paramètres, combien de temps prendrait la défragmentation (c'est à dire le compactage) de la moitié d'un disque de 16 GB ?

7. Certains dispositifs numériques grand public ont besoin de stocker des données, par exemple des fichiers. Indiquez un dispositif numérique récent qui nécessite l'enregistrement de fichiers et pour lequel une allocation contiguë serait une bonne idée.
8. De quelle manière MS-DOS implémente-t-il l'accès aléatoire aux fichiers ?
9. Considérons l'i-node de la figure ci-dessous. S'il contient 8 adresses directes de 4 octets chacune et si tous les blocs sont de 1024 octets, quelle est la taille maximale d'un fichier ?



10. Donnez un avantage des liens matériels par rapport aux liens symboliques et un avantage des liens symboliques vis-à-vis des liens matériels.
11. Le début d'une table de blocs libres juste après le formatage d'une partition disque ressemble à 100000000000000 (le premier bloc est utilisé par le répertoire racine). Le système recherche toujours les blocs libres à partir du bloc qui a le plus petit nombre ; ainsi après l'écriture du fichier A, qui requiert 6 blocs, la table des blocs libres est de la forme : 11111100000000. Donnez la table après chacune des opérations suivantes :
  - a) Le fichier B est écrit en utilisant 5 blocs.
  - b) Le fichier A est effacé.
  - c) Le fichier C est écrit en utilisant 8 blocs.
  - d) Le fichier B est effacé.
12. Que se passe-t-il si la liste des blocs libres ou la table sont perdues à la suite d'un plantage ? Existe-t-il un moyen de les reconstruire ou le contenu du disque est-il définitivement perdu ? Développez votre réponse pour le système de fichiers UNIX d'une part, et pour celui d'un système FAT-16 d'autre part.
13. Nous avons étudié en détail les sauvegardes incrémentales. Sous Windows, il est facile de savoir quand sauvegarder un fichier, car chaque fichier possède un bit d'archivage. Mais ce bit est absent sous UNIX. De quelle manière les programmes de sauvegarde d'UNIX ont-ils connaissance des fichiers à sauvegarder ?

14. Un système de fichiers UNIX a des blocs de 1 KiB et des adresses disque sur 4 octets. Quelle est la taille maximale d'un fichier si les i-nodes contiennent 10 entrées directes et une simple indirection, une double indirection et une triple indirection pour chaque fichier ?
15. Combien d'opérations disque sont nécessaires pour charger l'i-node du fichier `/usr/ast/courses/os/handout.t` ? Supposez que l'i-node du répertoire racine se trouve en mémoire, mais qu'aucun autre élément du chemin d'accès ne s'y trouve. Supposez aussi que tous les répertoires tiennent dans un bloc de disque.
16. Un RAID5 échoue si deux ou plus de ses disques tombent en panne dans un intervalle de temps limité. Disons que la probabilité qu'un disque tombe en panne à une heure donnée est de  $p$  ( $0 \leq p \leq 1$ ). Quelle est la probabilité qu'un RAID5 avec  $k$  disques tombe en panne à une heure donnée ?
17. Comparez les niveaux RAID de 0 jusqu'à 5 vis-à-vis des performances en lecture, des performances en écriture, de l'espace perdu et de la fiabilité.
18. Un fabricant de disques propose deux disques 5.25 pouces équipés de 10 000 cylindres. Le plus récent double la densité d'enregistrement linéaire de l'ancien. Quelles propriétés du disque sont meilleures sur le disque le plus récent et lesquelles sont identiques ?
19. Des requêtes de disque parviennent au pilote de disque pour les cylindres 10, 22, 20, 2, 40, 6 et 38, dans cet ordre. Un positionnement prend 6 ms par cylindre. Quel est le temps de positionnement nécessaire si l'on suit les méthodes suivantes :
  - a) Premier arrivé, premier servi.
  - b) Plus proche en premier.
  - c) L'algorithme de l'ascenseur (déplacement vers le haut en premier).

Dans tous les cas, le bras se trouve à l'origine au-dessus du cylindre 20.