

info201 : Système d'exploitation TD 1 : fichiers

Pierre Hyvernât
Laboratoire de mathématiques de l'université de Savoie
bâtiment Chablais, bureau 17, poste : 94 22
email : Pierre.Hyvernât@univ-smb.fr
www : <http://www.lama.univ-smb.fr/~hyvernât/>

Exercice 1 : bits et octets

Les préfixes usuels en informatique sont :

- l'octet, "1 o" pour une suite de 8 bits,
- le kibioctet, "1 kio" pour une suite de $2^{10} = 1024 \approx 10^3$ octets,
- le mébioctet, "1 Mio" pour une suite de 2^{10} kibioctets,
- le gibioctet, "1 Gio" pour une suite de 2^{10} mébioctets,
- le tébioctet, "1 Tio" pour une suite de 2^{10} gibioctets,
- ...

Question 1. Donnez le nombres d'octets dans 1Mio, 1Gio et 1Tio.

Question 2. Un kilooctet (1ko) fait *exactement* 1000 octets alors qu'un kibioctet fait *approximativement* 1000 octets. Un kibioctet a donc 2.4% d'octets en plus qu'un kilooctet.

De même, téraoctet (1To) fait *exactement* 10^{12} octets alors qu'un tébioctet fait *approximativement* 10^{12} octets.

Estimez le gain d'octets (en pourcentage) entre 1To et 1Tio.

Exercice 2 : questions de cours

Question 1. Qu'est-ce qu'un *fichier binaire* ? Donnez quelques exemples.

Qu'est-ce qu'un fichier texte *structuré* ? Donnez quelques exemples de formats de fichiers texte structurés.

Question 2. Que sont les *métadonnées* d'un fichier ? Donnez des exemples.

Est-ce que les métadonnées d'un fichiers occupent de l'espace disque ?

Question 3. Expliquez pourquoi les *données* d'un fichier de n octets occupent en général strictement plus que n octets sur son support de stockage.

Question 4. Donnez des avantages et inconvénients du découpage des fichiers en *blocs*.

Question 5. Si les blocs du système de fichiers font 4kio, quel est l'espace gaspillé (en pourcentage de la taille du fichier) sur un fichier de

- 1 octet,
- 1 kibioctet + 1 octet,
- 1 mébioctet + 1 octet,
- 1 gibioctet + 1 octet ?

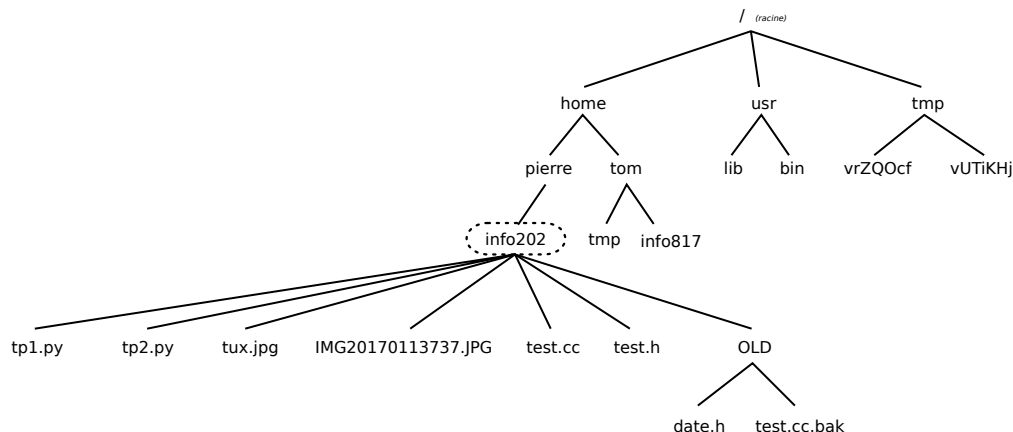
Question 6. Comment est il possible de référencer les blocs utilisés par un fichier ?

Essayer de détailler les avantages et inconvénients de votre (vos) méthode (méthodes).

Exercice 3 : noms de fichiers, chemins et motifs

Question 1. Qu'est-ce qu'un chemin d'accès ? Quelles différences y a t'il entre un chemin *absolu* et un chemin *relatif* ?

Question 2. Voici un morceau d'une arborescence de fichiers Unix : (**info201** est le répertoire courant)



Donnez des chemins absolus et relatifs pour accéder aux répertoires **OLD**, **tom**, **info201**, **vrZQOcf** et **tmp**.

Question 3. Pour chacun des motifs shell suivants, donnez la liste des fichiers correspondants dans l'arborescence ci dessus. (**info201** est le répertoire courant)

<code>*.py</code>	<code>*.?/*h</code>
<code>*.?</code>	<code>*.*</code>
<code>*[ch]</code>	<code>/*?*k</code>

Exercice 4 : pour aller plus loin

Question 1. Inventez et décrivez précisément un système de fichier !

Il s'agit de décrire comment stocker des fichiers sur un support de stockage vu comme un grand tableau d'octets. Essayer notamment de répondre aux questions suivantes :

- comment stocke t'on les données d'un fichier ?
- comment stocke t'on le nom et les métadonnées d'un fichier ?
- comment accède t'on aux données d'un fichier à partir de son chemin d'accès ?
- comment supprime t'on un fichier ?
- comment trouve t'on de la place pour un nouveau fichier ?

Question 2. Décrivez (en français ou par un schéma) l'état du support de stockage si votre système de fichiers contient l'arborescence **/tmp/projet** : contient uniquement la chaîne "TODO" **/tmp/a** : un fichier vide