

SYSTÈME

Devoir Surveillé n°1

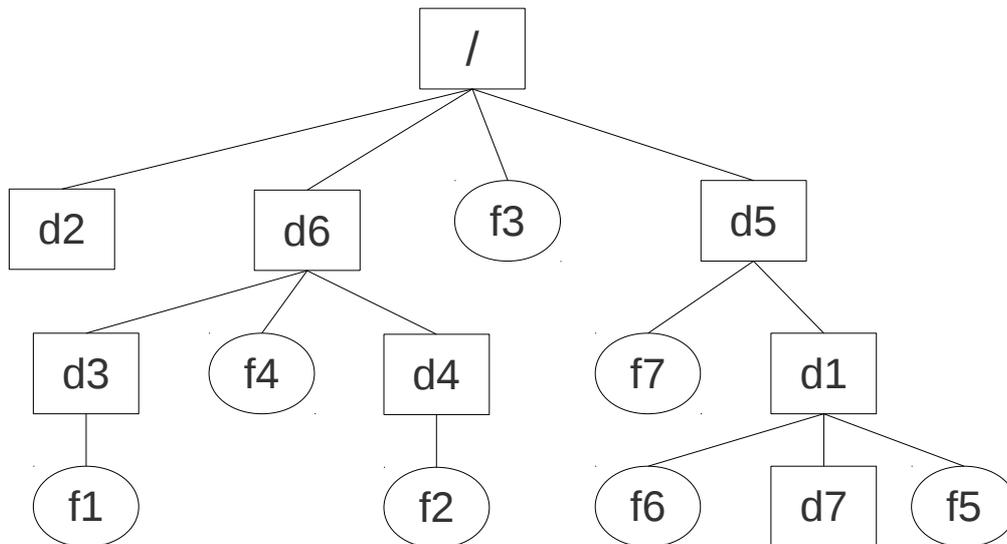
date : 19 octobre 2012, 15h45
 feuille A4 manuscrite personnelle autorisée.

durée : 1 heure
 barème : P1/10, P2/10.

NOM, Prénom :	Groupe :
----------------------	-----------------

Partie 1: Fichiers, répertoires et commandes (30 minutes, 10 points)

Voici un arbre de fichiers. Les ovales représentent des fichiers et les rectangles représentent des répertoires.



a) **Donnez le nom absolu de :**

d2	
f7	

b) **Vu de d4, donnez un nom complet relatif de :**

f4	
f1	
f2	
f5	

c) **On se place dans d1. En utilisant uniquement des noms complets relatifs, donnez une commande pour :**

supprimer f7	
lister d5	
copier f6 dans d4	

d) On se place dans le dossier d4

- après exécution de la commande : `mv ../../f3 ../.`

dans quel dossier se trouve le fichier f3 :

- après exécution de la commande : `cp ../f4 ./f8`

dans quel dossier se trouve le fichier f4 :

- après exécution de la commande : `ls ../..`

donner la liste des dossiers et fichiers affichés :

- après exécution de la commande : `rm -r /d5/f*`

donner la liste des dossiers et fichiers supprimés :

- e) Cette question est complètement indépendante des précédentes. Soit un répertoire où se trouvent les fichiers suivants :

`portable portion.exe porte.txt texte epart.c`

Quel(s) paramètre(s) le(s) plus court(s) possible, écrit(s) à l'aide du joker * de bash, faut-il fournir à la commande `rm` pour supprimer ensemble en une fois seulement certains des fichiers de cette liste, ceux qui sont indiqués ci-dessous, mais pas les autres ? Toutes ces questions sont indépendantes. Il n'y a pas d'ordre de difficulté croissante.

fichiers à supprimer avec un seul appel à <code>rm</code>	paramètre(s) le(s) plus court(s) de <code>rm</code>
<code>portion.exe texte</code>	
<code>portable portion.exe</code>	
<code>portable porte.txt portion.exe</code>	
<code>portable texte portion.exe</code>	
<code>epart.c porte.txt portion.exe</code>	

NOM, Prénom :

Groupe :

Partie 2: Commandes, redirections et tubes (30 minutes, 10 points)

Soit un fichier texte log.txt contenant des lignes telles que les suivantes (attention, c'est seulement un extrait du fichier, il contient bien plus de lignes, et les lignes peuvent être désordonnées) :

```
...
20120915.083712 modif essai.c      146
20120928.170545 ajout partie1.odt 2765
20121003.143020 suppr essai.c     0
20121015.134507 modif partie1.odt 2994
20121018.091023 modif partie1.odt 3278
...
```

Explications : Le but du fichier est de suivre l'évolution des fichiers dans un dossier. Chaque ligne représente une opération : ajout, suppression ou modification d'un fichier, à une date donnée. Au début de la ligne on trouve un nombre représentant la date et l'heure de l'opération, le format est AAAAMMJJ.HHMMSS. Par exemple, la date du DS d'aujourd'hui serait écrite 20121019.154500. Ce codage rend les dates peu lisibles, mais il permet de classer facilement les lignes dans l'ordre des dates. Après, on trouve le type d'opération (ajout, modif ou suppr), le nom du fichier concerné et la taille qu'il fait à l'issue de l'opération.

Donner les commandes complètes permettant de faire les opérations demandées. Les commandes sont toutes indépendantes, on repart à chaque fois de la situation initiale. Si vous n'arrivez pas à faire une question, passez à la suivante.

1) Recopier les lignes concernant les ajouts (le mot ajout en 2e colonne) du fichier log.txt dans un second fichier appelé ajouts.txt :

2) Pour chaque ligne de log.txt, afficher seulement la date et le nom du fichier concerné, ex :
20120915.083712 essai.c

3) Quelle commande permet de classer les lignes de log.txt sur les noms des fichiers ?

4) Écrire la commande qui enlève le point situé entre JJ et HH dans les dates :

5) Sachant que le premier nombre de chaque ligne représente une date, faire afficher les trois opérations les plus récentes du fichier log.txt, même au cas où il n'est pas forcément classé :

- 6) Chaque ligne de log.txt représente une opération sur un fichier. On voudrait savoir quel est le fichier concerné par le plus d'opérations au cours du mois de septembre 2012 : c'est à dire afficher le nom (seulement) de celui qui est le plus souvent mentionné durant ce mois.

- 7) Afficher la ligne du fichier ayant la plus petite taille présent dans log.txt pour un ajout ou une modification (pas les suppressions car la taille est zéro).

- 8) Faire afficher les opérations qui ont eu lieu du 15 septembre 2012 au 30 septembre 2012 :

- 9) Problème : afficher la liste des fichiers encore présents (selon log.txt). Ce sont des fichiers qui ont été ajoutés et qui n'apparaissent pas dans les suppressions. Il faut donc obtenir la liste des fichiers ajoutés et la liste des fichiers supprimés et garder uniquement ceux qui sont dans la première liste et pas dans la seconde. Écrire toutes les commandes nécessaires.

- 10) Admettons qu'on ait une commande appelée MiseAJour dont le rôle est de mettre le fichier log.txt à jour. Le problème c'est qu'elle met un certain temps à travailler pour détecter ce qui a changé parmi les fichiers. D'autre part, elle affiche des messages d'information sur la sortie des erreurs tout au long de son fonctionnement. Quelle ligne complète doit-on taper pour la lancer afin qu'elle nous laisse libre de lancer de nouvelles commandes et sans polluer l'affichage ?

- 11) Comment faire pour arrêter l'exécution de la commande MiseAJour de la question précédente au cas où celle-ci semble ne pas se terminer ?