

# L1 *Introduction aux systèmes d'exploitation*

## TD01

Amine NAJAH [amine.najahi@univ-perp.fr]

### EXERCICE 1 Découverte de l'interface graphique .....

À l'instar des systèmes d'exploitation Windows, les systèmes GNU/Linux offre à ses utilisateurs la possibilité d'utiliser une interface graphique. Il existe de nombreux systèmes GNU/Linux, appelés distributions, chacune ayant ses propres particularités destinées à différents type d'utilisation. Ubuntu est la distribution « grand public » qui connaît un grand succès de nos jours, et qui, est installée sur les machines de l'université. De plus, les systèmes GNU/Linux proposent différent type de gestionnaire de fenêtre... Le gestionnaire de fenêtre installé par défaut sur les machines de l'université se nomme « Gnome » :

1. Démarrez les ordinateurs en sélectionnant Ubuntu et observez ce qui ressemble à ce que vous connaissez de Windows.
2. Cherchez à ouvrir un gestionnaire de fichiers. Déplacez vous dans les dossiers...
3. Une fois le gestionnaire de fichier pris en main, identifiez votre dossier personnel et faites un clic-droit dessus puis observez les informations suivantes : propriétés, propriétaire, groupe, permissions, dates, tailles...

### EXERCICE 2 L'interface texte .....

Vous allez maintenant découvrir ce qui fait la force des systèmes GNU/Linux : l'interface texte ! Dans votre vie d'informaticien(ne), et particulièrement, dans votre parcours universitaire, vous allez être amené(e) à utiliser ces systèmes, les comprendre, les administrer... et ce, afin d'être efficace dans votre travail.

L'interface texte entre vous et la machine se fait à l'aide de commandes (notée ici entre [ ]) à entrer directement au clavier :

1. Dans le menu des applications, cherchez puis ouvrez un terminal. Avant toute chose, testez la commande [man]. À quoi sert-elle ? Testez la commande [man man].
2. Testez la commande [pwd]. À quoi sert-elle ? Quelle commande faut-il taper pour consulter sa page de manuel ?
3. Pour naviguer dans les répertoires, utilisez la commande [cd]. Il existe deux façons de naviguer dans les dossiers, soit en utilisant un chemin absolu, soit un chemin relatif. Expliquez la différence entre ces deux façons de se déplacer dans les dossiers.
4. Avec la commande [cd], comment retourner rapidement au dossier personnel ?
5. Qu'est ce que l'arborescence de fichier ? Examinez là.
6. Déplacez vous dans /bin ; de là, rendez vous en /usr/bin ; revenez à votre répertoire personnel ; de là, allez au répertoire personnel de votre voisin. Vérifiez votre position.
7. Enfin, pour ne plus avoir à aller chercher un terminal dans le menu des applications, ajoutez en un au lanceur.

### EXERCICE 3 La structure de l'arborescence (les références . et ..) .....

1. Testez la commande `[ls]`. Quel effet a cette commande? La commande `[ls]` peut prendre des options en paramètre. Testez la commande avec l'option suivante `[ls -a]`. Quelle différence voyez vous avec et sans cette option?
2. Testez la commande `[ls -d .]`. Quel est son effet? Comparez là avec la commande suivante : `[ls .]`.
3. Que font `[ls -d ..]` et `[ls ..]`?
4. Que fait `[ls -i]`?
5. Comparez l'inode du répertoire . (`[ls -id .]`) à celui du répertoire personnel (`[ls -id ../moi]`).
6. À partir des références . et des commandes nécessaires, construisez l'arbre de l'arborescence des fichiers jusqu'à atteindre la racine. À quoi sert la référence ..?
7. Trouvez l'inode de la racine (`[ls -id /]`), et du père de la racine (`[cd /]` puis `[ls -id ..]`).
8. Dessinez un arbre représentant `/`, `/home/moi`, `/bin`, `/usr/bin` avec leurs numéros d'inodes et les références . et .. (chaque répertoire contient la liste -non exhaustive- de ses fichiers, avec pour chacun son nom et son inode).
9. Naviguez à la main pour montrer l'utilité de .. (on ne peut pas remonter directement parce qu'on ne connaît pas le nom du père mais seulement son inode et le nom générique ..). Pourquoi ne pas avoir mis le nom du père plutôt que ..?
10. À quoi sert la référence .? Avec le dessin de l'arbre précédent, trouver l'inode du répertoire courant avec et sans la référence . .
11. Finalement, à quoi sert l'inode et que contient-il?

### EXERCICE 4 La structure de l'arborescence (les liens) .....

1. Testez et observez le résultat de la commande `[ls -l]`. Décrivez l'ensemble des informations affichées pour chaque fichier.
2. Combien de liens un fichier ordinaire compte t-il?
3. Le nombre de liens d'un répertoire est-il constant?
4. Le nombre de liens d'un répertoire peut-il être inférieur à 2? Comptez le nombre de répertoires inclus dans un répertoire et comparez le à son nombre de liens.
5. Comptez les liens de `/` avec la commande `[ls -ld /]`? Vérifiez tous les endroits où l'inode 2 est présent.