

1 Découverte de l’environnement Unix

Se connecter aux machines

Pour se connecter, il y a deux étapes , et vous avez besoin d’informations données sur votre certificat de scolarité :

1. Composez et validez votre *login*. Il s’agit de votre identifiant utilisateur, qui est l’initiale de votre nom, en minuscule, suivie de votre numéro de carte d’étudiant : par exemple c140154.
2. Composez et validez à l’aveugle votre *password*. Ce mot de passe est par défaut votre numéro INE en lettres minuscules, également inscrit sur votre carte d’étudiant ou votre certificat de scolarité. (si vous avez changé votre mot de passe initial, vous pouvez le retrouver en allant sur <https://sesame.univ-amu.fr/>)

Une fois connecté, vous arrivez sur votre bureau. Ouvrez votre dossier personnel. Vous y trouvez quelques répertoires (Bureau, Documents,...). Si vous avez un document "IMPORTANT-A-LIRE", lisez-le attentivement, il vous donne des explications sur votre compte utilisateur.

Ouvrir un terminal de commande

Ouvrez une fenêtre de terminal de commandes : celle-ci est accessible via *Applications - Système - Terminal* ou *Applications - Accessoires - Terminal*.

Le terminal indique qu’il est prêt à recevoir la saisie d’une commande au clavier en affichant *l’invite de commande* ou prompt : il s’agit d’une chaîne de caractères affichée en début de ligne, et qui prend généralement par défaut l’aspect général vosre-login@serveur-execution:répertoire-courant\$. Le répertoire courant (ou *current directory*) est le dossier dans lequel vous vous trouvez. Ce dossier est au départ votre *répertoire-maison*, c’est-à-dire le lieu de stockage de votre compte personnel.

Commandes à connaître impérativement

Au cours de ce TP, vous devez vous familiariser avec les commandes suivantes :

- Afficher le nom du répertoire courant : pwd (Print Working Directory).
- Afficher le contenu d’un répertoire : ls (LiSting).
- Se déplacer, changer de répertoire courant : cd (Change Directory).
- Créer de nouveaux répertoires : mkdir (MaKe DIRectory).
- Copier des fichiers et des répertoires : cp (CoPy).
- Renommer et déplacer des fichiers et des répertoires : mv (MoVe).
- Supprimer des fichiers et des répertoires : rm et rmdir (ReMove et DIRectory).

Ainsi que les commandes :

- Compiler un fichier source en C : gcc (Gnu C Compiler).
- Aide pour les commandes Unix et les fonctions C : man (MANual).

Et enfin les commandes :

- Navigateur web : par exemple konqueror ou firefox ou iceweasel
- Editeur de texte pour programmer : gedit ou geany

Ces dernières commandes ne se terminant pas instantanément, on souhaite presque toujours les lancer en tâche de fond afin de récupérer l’invite de commande du terminal avant la fin de l’exécution de la commande. Pour cela, il faut faire suivre le nom de la commande par une esperluette (&), par exemple

konqueror & ou gedit &. Si l'on a oublié l'esperluette et que l'on souhaite récupérer l'accès terminal, on peut interrompre l'exécution du programme avec la combinaison de touches [CTRL+C].

Scénario de commandes à exécuter

1. Assurez vous en tapant la commande `pwd` (ou au simple regard de votre prompt si celui-ci est correctement configuré) que vous êtes bien dans votre répertoire-maison : la chaîne affichée doit se terminer par `/votre-login`. Si ce n'est pas le cas, c'est que vous êtes ailleurs, et tapez la commande `cd` (ou encore `cd ~`) qui permet de revenir dans votre répertoire maison.
2. Tapez la commande `ls`, qui permet de lister le contenu du répertoire courant : vous y trouvez `Bureau`, `Documents`, etc.
3. Créez un répertoire nommé `intro` avec la commande `mkdir intro`. Tapez `ls` afin de vérifier que le répertoire est bien créé. A l'intérieur de celui-ci, créez 3 sous-répertoires nommés `tp1`, `tp2`, `tp3` avec des commandes suivant le modèle `mkdir intro/tp1`.
4. Afin d'éviter les répétitions de frappes et les erreurs de frappes, vous disposez de l'historique des dernières commandes avec les flèches HAUT/BAS, ainsi que de la complétion automatique des noms de fichiers avec la touche de TAB.
5. Vérifiez que les 3 sous répertoires dans `intro` ont bien été créés, avec la commande `ls intro`, ou avec la commande `ls -R` qui affiche *récurivement* toute une hiérarchie dans un répertoire.
6. Allez dans le répertoire `intro` avec la commande `cd intro`. Vérifiez où vous êtes avec `pwd` ou le prompt.
Déplacez-vous ensuite dans `tp1` avec la commande `cd tp1`. Vous pouvez revenir un cran plus haut dans la hiérarchie, c'est-à-dire dans `intro` avec la commande `cd ..` (*deux points accolés* dans un chemin indiquent le répertoire parent).
7. Grâce à plusieurs déplacements élémentaires avec `cd`, vérifiez que vous êtes capable de revenir à n'importe quel point d'arrivée à partir de n'importe quel point de départ parmi les 5 points que nous connaissons, à savoir :
`~`, `~/intro`, `~/intro/tp1`, `~/intro/tp2`, `~/intro/tp3`.
8. Allez dans `~`. Vérifiez que vous pouvez aller dans `~/intro/tp3` en une seule étape avec `cd ~/intro/tp3`. Une fois dans ce répertoire, allez directement dans `~/intro/tp2` grâce à la commande `cd ../tp2`, puis dans `~/intro/tp1` en tapant `cd ../tp1`.
9. Avec l'enchaînement de commandes de votre choix, créez 3 répertoires de telle sorte qu'on ait la hiérarchie `~/intro/tp1/download`, `~/intro/tp1/images`, `~/intro/tp1/paroles`.
10. Lancez un navigateur web et un éditeur de texte avec les commandes konqueror & et gedit &. N'oubliez pas l'esperluette finale pour les lancer en tâches de fond et ainsi récupérer immédiatement la main sur le terminal. Allez à l'adresse <http://www.google.fr>, et ajoutez un signet.
11. Recherchez dans Google les paroles de *It's a Sin* des Pet Shop Boys et copiez les dans l'éditeur de texte. Sauvez le document sous le nom `it-s-a-sin.pet-shop-boys.txt` dans le répertoire `~/intro/tp1/download`.
12. Faites de même avec des chansons du même groupe et d'autres groupes, et sauvegardez les dans le même répertoire avec le même modèle de noms : `always-in-my-mind.pet-shop-boys.txt`, `can-you-forgive-her.pet-shop-boys.txt`, `high-hopes.pink-floyds.txt`, `smooth-operator.sade.txt`, `no-ordinary-love.sade.txt` .

13. Trouvez également sous Google Images, des photos au format JPG de ces différents groupes et sauvez les dans `~/intro/tp1/download` sous les noms `pet-shop-boys.jpg`, `sade.jpg`, `pink-floyds.jpg`. Vérifiez que tout ce petit monde est bien là en listant le contenu de `~/intro/tp1/download`.
14. Visualisez vos images avec `xv` ou `display`. Par exemple : `xv sade.jpg &`.
15. Allez dans `~/intro/tp1/download` et copiez le fichier `sade.jpg` sous un nom différent `sade-adu.jpg` dans ce même répertoire avec `cp sade.jpg sade-adu.jpg`.
16. Recopiez `sade.jpg` mais cette fois vers d'autres répertoires : dans `~/intro/tp1` avec `cp sade.jpg ..`, et dans `~/intro/tp1/images` avec `cp sade.jpg ../images`.
17. Vérifiez avec `ls` que les copies sont bien au bon endroit, puis supprimez les avec la commande `rm` : Si vous êtes toujours dans `~/intro/tp1/download`, avec les commandes `rm sade-adu.jpg`, `rm ../sade.jpg` et `rm ../images/sade.jpg`.
18. Allez dans `~/intro/tp1/download`, et déplacez `it-s-a-sin.pet-shop-boys.txt` du répertoire de téléchargement vers le répertoire de paroles avec par exemple `mv it* ../paroles`. Vérifiez avec `ls` que le fichier n'est plus dans `~/tp1/download` mais dans `~/tp1/paroles`.
19. Allez dans `~/intro/tp1/paroles`, et déplacez toutes les autres chansons vers ce même répertoire avec par exemple `mv ../download/*.txt .` (le point simple (.) indique ici le répertoire courant de la même façon que 2 points accolés (..) indiquent le répertoire père).
20. Allez dans `~/intro/tp1`. Renommez le répertoire de paroles sous le nom `lyrics` avec la commande `mv paroles lyrics`. Vérifiez le résultat avec `ls`.
21. Déplacez toutes les images de `~/intro/tp1/download` vers `~/intro/tp1/images`. Vérifiez avec `ls` que le déplacement s'est bien déroulé.
22. Allez dans le répertoire `~/intro`. Déplacez tout le répertoire `~/intro/tp1/images` vers `~/intro/tp2` avec la commande `mv tp1/images tp2`. Vérifiez avec `ls` que `images` est bien arrivé dans `tp2` et a bien disparu de `tp1`.
23. Allez dans le répertoire `~/intro`. Faites une copie du répertoire `~/intro/tp2/images` dans `~/intro/tp3/images` avec la commande `cp -r tp2/images tp3`. Vérifiez avec `ls` qu'un exemple de `images` est bien à la fois dans `tp2` et `tp3`.
24. Allez dans le répertoire `~/intro`. Effacez toutes les images de `~/intro/tp3/images` avec `rm tp3/images/*`. Vérifiez que le répertoire `images` est toujours là mais vide.
25. Allez dans le répertoire `~/intro`.
Avec `rmdir tp3/images` effacez le répertoire `~/intro/tp3/images` qui maintenant est vide.
26. Allez dans le répertoire `~/intro`.
Avec `rmdir tp3` effacez le répertoire `~/intro/tp3` qui maintenant est vide.
27. Essayez la même chose sur le répertoire non-vide `~/intro/tp2`. Vérifiez que la commande échoue. Effacez maintenant directement ce répertoire et tout son contenu avec la commande `rm -r tp2` (effacer récursivement).
28. Pour toute aide sur une commande et ses options, utiliser `man` : Par exemple, `man ls` ou `man cp`. Vous obtiendrez également cette documentation sur le web en tapant ces mêmes mots dans Google.

2 Premiers pas en Python

2.1 Premier programme en Python

```
a = 2
b = 3 * a
c = a + b
print("bonjour:" , a , "+" , b , "=", c)
```

1. Créez un répertoire `~/intro/tp1/prog`. Allez dedans, ouvrez `gedit` & créez un fichier `~/intro/tp1/prog/bonjour.py` qui contient le programme ci-dessus.
2. Exécuter le programme en invoquant Python suivi du nom du fichier : `python3 bonjour.py`

2.2 Entrées/sorties simples avec `input` et `print`

Dans la plupart des programmes que vous ferez, vous voudrez que :

- l'utilisateur puisse fournir des données au programme (les entrées);
- le programme affiche des résultats (les sorties).

Une manière simple de faire des entrées/sorties est d'utiliser les fonctions `input` et `print`, qui fonctionnent de la manière qui suit.

Si `i` est une variable associée à une valeur du type `int`,

- pour lire une valeur entière au clavier et la stocker dans `i`, il suffit de faire :
`i=int(input())`
- pour afficher la valeur de `i`, il suffit de faire :
`print("La_valeur_de_i_est_i={}".format(i))`

Si `r` est une variable associée à une valeur du type `float`,

- pour lire une valeur réelle au clavier et la stocker dans `r`, il suffit de faire :
`r=float(input())`
- pour afficher la valeur de `r`, il suffit de faire :
`print("La_valeur_de_r_est_r={}".format(r))`

2.3 Arithmétique

Ecrire un programme qui

- lit au clavier 2 nombres entiers `a` et `b` et affiche à l'écran les valeurs qui ont été lues,
- calcule et affiche la somme et le produit de `a` et `b`,
- calcule et affiche le résultat de la division euclidienne de `a` par `b` (lorsque le calcul est possible), c'est-à-dire leur quotient entier et le reste de la division.

2.4 Conversion

Ecrire un programme qui convertit en heures, minutes et secondes une durée exprimée en secondes, donnée par l'utilisateur.