

Compte rendu TP5

PSE

Substitution de descripteurs de fichiers

Communication entre processus par tubes

Exo 1 : Substitution de descripteurs de fichiers

Principe : On souhaite créer un fichier en mode écriture et faire en sorte qu'il devienne la sortie standard, et exécuter la commande **ls -l** dont le résultat devrait s'afficher dans le fichier créé

Test :

```
[0]utilisateur@Ubuntu:~$ python3 TP5exo1.py f.txt
[0]utilisateur@Ubuntu:~$ cat f.txt
total 56
drwxr-xr-x 2 utilisateur utilisateur 4096 juil.  4 02:37 Bureau
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Documents
-rwxrwxr-x 1 utilisateur utilisateur    0 nov.  10 08:32 f.txt
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Images
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Modèles
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Musique
drwxr-xr-x 4 root          root        4096 juin   1 2016 pipework
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Public
drwxr-xr-x 2 utilisateur utilisateur 4096 juil.  4 02:47 Téléchargements
-rw-rw-r-- 1 utilisateur utilisateur  136 nov.  10 08:32 TP5exo1.py
-rw-rw-r-- 1 utilisateur utilisateur  134 nov.  10 08:31 TP5exo1.py~
drwxr-xr-x 2 utilisateur utilisateur 4096 juin   1 2016 Vidéos
-rw-rw-r-- 1 utilisateur utilisateur 4206 juin  11 2016 vmware-xdg-detect-de
drwxrwxr-x 3 utilisateur utilisateur 4096 juin   1 2016 workspace
```

Le résultat de la commande **ls -l** se trouve bien dans le fichier **f.txt** qui est créé.

Exo 2 : Substitution de descripteurs de fichiers – bis

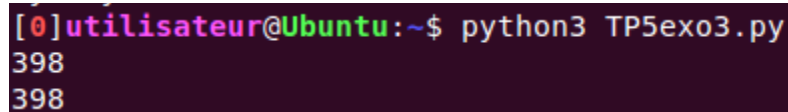
Principe : On souhaite créer un fichier temporaire qui va devenir notre sortie d'erreurs. Utiliser la fonction `traceback.print_stack()` qui écrit la trace d'appel du programme dans la sortie d'erreurs qui sera normalement notre fichier temporaire. Puis après vérification de son contenu, remettre la sortie d'erreurs dans son état par défaut.

Test : Lors de l'exécution de mon programme rien ne s'affichait. Je ne pense pas que mon code soit faux.

Exo 3 : Communication par tube anonyme

Principe : On souhaite créer un tube de communication entre un processus père et son processus fils. Le processus père (coté écriture du tube de communication) devra alors communiquer un nombre aléatoire entre 0 et 1000 à son processus fils (côté lecture du tube de communication) .

Test :

A terminal window with a dark background. The prompt is [0]utilisateur@Ubuntu:~\$. The command python3 TP5exo3.py has been executed. The output shows the number 398 on two separate lines.

```
[0]utilisateur@Ubuntu:~$ python3 TP5exo3.py
398
398
```

Le résultat obtenu est celui attendu, le nombre généré est 398, celui lu est bien 398.