



Polytechnique Montréal

Département de Génie Informatique et Génie Logiciel

INF2610 - Noyau d'un système d'exploitation

TP 2

Communication

Question	Description	Pointage
1	communication par tubes nommés	8
2	Construction d'une ligne de commande	8

Structure du TP

Ce travail est composé de deux questions principales visant à vous familiariser avec le concept de communication inter-processus.

Il vous est toujours possible de regarder si vous avez réussi ou non une question en exécutant le script `./evaluate.sh` situé à la racine du dossier du TP.

À la fin du TP, vous devrez lancer la commande `make handin` dans le répertoire principal du tp afin de créer l'archive `handin.tar.gz` que vous devrez remettre sur autolab.

Prenez le temps de lire attentivement l'énoncé.
BON TP! :)

Communication par tubes nommés (8)

L'objectif de cette question est de faire communiquer deux programmes à l'aide des tubes nommés.

Le code à compléter se trouve dans le fichier **TubesNummes.c**.

Vous devez faire communiquer les programmes `johnDoe` et `uncharted` pour qu'ils puissent vérifier un échange de jetons d'authentification. Vous devrez respecter l'échange suivant:

- Le jeton `b061658b83ab74f6` devra être envoyé dans le tube `/tmp/pipe_a`
- Le jeton `84613ab6d3076bcd` devra être envoyé dans le tube `/tmp/pipe_b`

ATTENTION - Chaque programme prend comme UNIQUE paramètre le PID du processus parent (après le nom du programme, par convention).

Cela dit, vous remarquerez probablement déjà que vous avez une communication bidirectionnelle. Vous aurez donc besoin de deux tubes pour établir une communication efficace. Vous devrez utiliser 2 tubes nommés que vous devrez créer sous `/tmp/pipe_a` et `/tmp/pipe_b`.

ATTENTION Vous devrez créer 4 processus fils donc les tâches seront les suivantes:

- Un fils sera responsable de lancer johnDoe
- Un fils sera responsable de lancer uncharted
- Un fils sera responsable d'envoyer le jeton d'authentification dans le tube /tmp/pipe_a
- Un fils sera responsable d'envoyer le jeton d'authentification dans le tube /tmp/pipe_b

johnDoe devra écrire dans le tube /tmp/pipe_a. l'inverse, uncharted devra écrire dans le tube /tmp/pipe_b. Si tout se passe bien, vous recevrez la note de 8/8.

Reproduction d'une ligne de commande bash (/8)

L'objectif de cette question est de reproduire une ligne de commande *bash* avec du code C.

Le code à compléter se trouve dans le fichier **LigneDeCommande.c**.

Vous devrez reproduire la ligne de commande suivante, mettant en relation les trois programmes *prog1*, *prog2* et *prog3*:

```
./prog1 2>&1 | ./prog2 > tmpfile; ./prog3
```

Chaque programme peut écrire une lettre sur la sortie d'erreur standard ou la sortie standard et/ou demander une certaine entrée standard. Portez une attention particulière aux redirections bash vues en classe.

Voici quelques astuces pour vous aider à debug et réaliser votre programme:

- Le fichier *tmpfile* se trouve sous *./executables/Question2/tmpfile*
- Si jamais au lancement de *./grade.sh* vous obtenez une erreur python mentionnant un *unexpected byte*, c'est qu'il y a un problème avec une redirection de pipe et/ou que les deux programmes utilisant le pipe ne sont pas exécutés.
- Dans le cas où vous voulez savoir quel input/output est attendu et/ou reçu, il est possible de consulter les fichiers de logs des programmes en commentant les lignes 14-16 de *grade.sh*. Vous devez ensuite supprimer manuellement les fichiers et d/commenter les lignes pour retrouver votre score.