

INF2610

TP #4 : Communication interprocessus

Groupe 01L

Polytechnique Montréal

Automne 2023

Date de remise: Voir le site Moodle du cours

Pondération: 5%

Objectifs

Ce travail pratique (TP) a pour but de vous familiariser avec les tubes anonymes et nommés, utilisés pour faire communiquer des processus. À l'issue du TP, vous serez capable de:

- faire communiquer des processus au moyens de tubes de communication anonymes et nommés, et
- rediriger les flux d'entrée et de sortie d'un processus vers des fichiers et des tubes de communication.

Ce TP est composé de deux questions indépendantes que vous pouvez traiter séparément.

Compilation, exécution et remise

Pour compiler et exécuter le TP, lancez successivement les commandes `make` et `./CommLab` dans le répertoire du TP.

Lancez la commande `make handin` dans le répertoire du TP afin de créer l'archive `handin.tar.gz`. Déposez l'archive sur le site Moodle du cours, après avoir ajouté au nom du fichier `handin` vos matricules. Attention, vérifiez bien que tous les fichiers nécessaires à l'évaluation de votre travail sont bien inclus dans le fichier soumis.

Barème

Question	Description	Points
1	Création et transformation des processus	4
	Tubes et redirections des E/S standards	6
	Fermeture des descripteurs de fichiers et attente de fin des processus	2
	Résultat correct	2
2	Communication par tubes nommés	4
	Clarté du code et commentaires	2

Question 1 - Tubes anonymes et redirections

L'objectif de cette question est de reproduire en langage C le traitement réalisé par la composition de commandes suivante:

```
cat In.txt | tr [a-z] [A-Z] | diff - In.txt > Output.txt
```

Où:

- `In.txt` est un fichier fourni contenant le text à inverser;
- `Out.txt` est un fichier que vous devez créer, au moment voulu dans le programme, en réalisant l'appel suivant:

```
open("Out.txt", O_WRONLY | O_CREAT | O_TRUNC, 0660);
```

- `tr` et `diff` sont des fichiers exécutables dont les chemins d'accès sont dans la variable d'environnement `PATH`. La commande `tr [a-z] [A-Z]` 'traduit' tout les lettres minuscules pour quelles deviennent majuscules. Le résultat de cette opération est affiché sur la sortie standard. Si le fichier à traduire n'est pas précisé, la commande `tr` va traduire les caractères de chaque ligne lue à partir de son entrée standard. Le résultat de cette traduction est affiché sur la sortie standard. La commande `diff - In.txt > Output.txt` se charge de comparer les fichiers `In.txt` et l'entrée standard. Le résultat de cette comparaison est affiché dans le fichier `Output.txt`. Dans le cas de la composition de commandes ci-dessus, les fichiers `In.txt` et l'entrée standard devraient être différents.
- Les opérateurs "`<`" et "`>`" permettent respectivement de rediriger l'entrée et la sortie standards vers des fichiers.

📁 Complétez le code dans le fichier ***TubesAnonymes.c*** afin de réaliser le même traitement que la ligne de commandes ci-dessus. Vous devez également prendre en compte les requis suivants:

- Les trois processus *P1*, *P2* et *P3* que vous allez créer pour exécuter respectivement les commandes simples `cat In.txt`, `tr [a-z] [A-Z]` et `diff - In.txt > Output.txt` doivent obéir à la hiérarchie suivante. Le processus *P3* est créé par le processus principal alors que le processus *P1* et *P2* sont créés par le processus *P2* et *P3* respectivement.
- Les descripteurs de fichiers non utiles doivent être fermés.
- Chaque processus père qui ne se transforme pas doit attendre la fin de ses fils avant de se terminer.
- Il n'est pas demandé de traiter les erreurs. Par contre, en cas de doute sur le bon fonctionnement de votre programme, n'hésitez pas à tester les valeurs de retour de vos appels système.

Question 2 - Tubes nommés et redirections

Dans cette question, vous devez reproduire le même traitement mais en utilisant un tube nommé au lieu d'un tube anonyme:

```
cat In.txt | tr [a-z] [A-Z] | diff - In.txt > Output.txt
```

Vous devez remplacer le fichier `Out.txt` par `Outn.txt`. Le code à compléter se trouve dans le fichier ***TubesNommes.c***.