Séance de TP N°5
05.tubes – Manipulation de tubes

Créez votre répertoire de TP *~/l3i6/05.tubes* et placez-vous dessus. Récupérez dans votre répertoire de TP le fichier *Fourniture05.tgz* sur le site de cet enseignement[1] et déballez-le. Ce répertoire contient :

- ***Makefile.gen*** et le script ***configure*** pour produire un ***Makefile*** adapté à votre système.
- pageur1.c, upper2lower.c, popen.c.

Utilisation d'un tube par plusieurs processus

Écrivez un programme *tubeP2F.c* qui crée un tube et 2 processus fils, appelés *fils1, fils2*. Les processus *fils1* et *fils2* écrivent 100 fois chacun dans le tube un message formé respectivement de 50 chiffres "1" ou "2", précédés du numéro de ligne sur 3 chiffres et d'un espace.
Le processus *père* lit les messages et les affiche sur la voie standard de sortie. Plusieurs scénarios de synchronisation sont possibles entre *fils1* et *fils2*:

- aucune synchronisation : dans ce cas, plusieurs exécutions successives devraient donner des entrelacements différents des messages. Noter cependant, que puisque les messages sont plus petits que la taille du tube (moins de 4096 caractères), chaque message est écrit de manière atomique, sans contenir les caractères de l'autre.
- une synchronisation (qui sera réalisée lors d'une séance de TP suivante) pour garantir l'alternance stricte *fils1, fils2, fils1*...

Testez ce programme sans synchronisation.

Utilisation de *popen* en écriture

Étudiez et compiler le programme ***pageur1.c*** (vu en cours). Exécutez ce programme en donnant, comme fichier de texte à lire, le nom du fichier source précédent en argument. Vérifiez que le programme ***pageur1*** envoie bien le contenu du fichier donné en argument au programme pageur par défaut pour l'affichage.

Exportez une variable d'environnement *PAGER* contenant le nom d'un programme de type "pageur", par exemple ***less***:

```
$ export PAGER=less
```

Réexécutez le programme pageur1 avec le fichier pageur1.c et vérifiez le comportement. Changez de programme pageur en modifiant le contenu de la variable *PAGEUR*, par exemple la commande ***wc*** et recommencez l'exécution.

Écrivez un programme ***pageur2.c*** qui réalise le même comportement que le programme ***pageur1.c***, mais en utilisant les primitives ***popen***/***pclose***. Affichez à la fin le code de retour du programme appelé par ***popen***.

[1] http://deptinfo.unice.fr/~lahire/enseignement/SYSL3

Utilisation de *popen* en lecture

Le programme ***upper2lower*** fourni est un filtre qui convertit les lettres majuscules lues sur la voie *stdin* en lettres minuscules sur la voie *stdout*. Testez ce programme avec le fichier ***Makefile*** :

```
$ cat Makefile | less
$ cat Makefile | upper2lower | less
```

Écrire un programme ***adaptateur.c*** qui crée une liaison par **popen** en lecture avec le programme ***upper2lower*** fourni. ***adaptateur.c*** affiche une invite, puis entre dans une boucle de lecture de lignes sur l'extrémité du tube jusqu'à la fin de fichier.
À chaque pas de l'itération, le programme affichera la ligne lue et réaffichera l'invite. On constatera l'adaptation réalisée par le processus fils associé (ici de traduire les majuscules en minuscules).
Après la détection de la fin de fichier, le programme attendra son fils par ***pclose*** et récupérera son statut qui sera affiché en hexadécimal. Testez ce programme.

Pour les plus courageux ...

Etudiez le code source du module \texttt{popen.c} qui fournit une implémentation des fonctions de bibliothèque ***popen(3)*** et ***pclose(3)***.

Modifiez ce module pour que les fonctions ***popen(3)*** et ***pclose(3)*** acceptent la valeur "e" de l'argument ***type***, pour réaliser une adaptation des messages écrits sur la voie standard d'erreur par la commande exécutée.

Testez ce module avec un programme adaptateur ***gcc2***, variante du programme de la version précédente, pour adapter les messages d'erreur du compilateur ***gcc*** standard. L'adaptation consistera à écrire les numéros de ligne d'erreur avant le nom du fichier.

Cette expérience montre qu'il est possible avec ***popen***/***pclose*** de réaliser des adaptations de comportement de programmes sans avoir accès aux fichiers sources, simplement en filtrant les voies d'E/S de ces programmes.