

TP :

Client/Serveur avec des tubes

Première année d'IUT Informatique

Description de l'application

Nous allons construire une application Client/Serveur en utilisant des tubes.

Le schéma de l'application est assez simple : un seul *serveur* accepte et traite les requêtes transmises par différents clients. Après traitement, le serveur envoie les résultats aux *clients*. Il faut permettre à plusieurs clients d'envoyer des requêtes au serveur.

Le service réalisé par le serveur sera un service d'évaluation d'expressions arithmétiques. Par exemple, si le client envoie la requête “ $(1+4)*5$ ”, le serveur évalue cette expression et retourne le résultat 25 au client concerné.

Nous allons utiliser les différents pipes vus en cours pour réaliser notre application :

- Des tubes nommés pour établir la communication entre les clients et le serveur. Il faudra un tube pour établir la communication avec le serveur (sens client vers serveur) ; et un tube par client pour que le serveur puisse retourner le résultat de l'évaluation. Cela veut dire, que la requête d'un client doit contenir un identifiant qui permet de retrouver facilement le tube nommé qui lui est associé. Par exemple, on peut utiliser la structure suivante pour représenter une requête :

```
#define BUFFER_SIZE 20
struct requete_client_serveur {
    pid_t client_pid; //PID du Client
    char expression[BUFFER_SIZE - 1]; //text de l expression a evaluer.
};
```

Par convention, le tube nommé qui correspond à un client de PID n sera : `/tmp/client_n_fifo`. La fifo du serveur sera : `/tmp/serveur_fifo`

- Pour évaluer les expressions arithmétiques le serveur pourra utiliser un programme externe (par exemple `bc`). Dans ce cas, on utilisera un tube processus pour établir la communication avec ce programme externe.