

# Systeme TD 5

## EXERCICE 1 : mise en œuvre du cours

---

Reprendre les éléments serveur et client du cours numéro 5 (sockets) pour écrire et tester deux programmes :

1. Serveur.c : programme qui attend une connexion TCP sur le port 6000. Lorsque la connexion est établie, le serveur affiche "**client connecté**". Puis le serveur se met en attente de réception (d'un message) sur la socket. Le serveur reçoit le message du client et l'affiche. Le serveur envoie alors un message au client (par la socket). Le serveur ferme sa socket de communication, puis ferme la socket d'écoute.
2. Client.c : programme qui se connecte au serveur sur le port 6000. Il affichera "**connexion ok**". Le client enverra un message au serveur, puis se mettra en attente de la réponse du serveur. Lorsque le message arrive, le client l'affiche puis ferme la connexion.

On pourra, pour cet exercice, utiliser pour le serveur l'adresse **127.0.0.1** et lancer les deux programmes sur la même machine, mais dans deux **terminaux différents**.

## EXERCICE 2

---

Modifier le programme précédent pour faire une discussion à deux. Chaque côté, tour à tour, saisit et envoie un message sur le réseau.

Les faire échanger 3 messages chacun ou mettre en place une discussion infinie avec possibilité d'interrompre le dialogue (détection d'une chaîne de caractères particulière).

Pour cet exercice, on pourra faire l'essai avec un voisin sur le même réseau IP.

## EXERCICE 3 : serveur permettant des connexions successives

---

Modifier le serveur de l'exercice précédent pour qu'il accepte **successivement** plusieurs connexions de la part de clients. On fera afficher par le serveur l'adresse IP et le port du client qui se connecte.

## EXERCICE 4 : serveur de temps

---

Faire deux programmes (serveur et client) permettant au client de se connecter au serveur pour obtenir l'heure de ce dernier. C'est-à-dire qu'un client se connecte et le serveur lui envoie sa date et son heure, puis se déconnecte. Il pourra accepter **successivement** plusieurs connexions.

## EXERCICE 5 : serveur avec connexions simultanées

---

Modifier le serveur de l'exercice 3 afin qu'il accepte **simultanément** plusieurs connexions de la part de clients.

On fera simplement afficher au serveur l'adresse IP du client qui se connecte et les messages que le client lui envoie. On utilisera un principe de *forking* pour gérer les différents clients.