

---

---

## IF4 DIST Sockets

### TP1 et TP2

*T. Grandpierre*

---

---

### ***Communications en mode non connecté***

#### **Exercice 1**

Ecrire deux programmes, un client et un serveur :

Le client reçoit en argument sur la ligne de commande :

- Le nom de la machine hébergeant le serveur
- Le port de ce serveur
- Une chaîne de caractères

Le serveur reçoit en argument sur la ligne de commande :

- Le port sur lequel il va attendre les connexions des clients

#### **Objectifs :**

Le client envoie vers le serveur :

- Son numéro de PID
- La chaîne de caractères qui lui a été passée en argument

Le serveur :

- reçoit ces informations,
- les affiche
- retourne au client son numéro de PID ainsi que la chaîne reçue.

Le client affiche le numéro de PID reçue du serveur et la chaîne de caractères

Note :

La fonction "*gethostbyname*" permet de récupérer l'adresse d'une machine dont le nom est donné en argument.

### **Communications en mode connecté**

#### ***Exercice 2***

Ecrire un serveur qui attend une requête de connexion sur son port d'écoute et qui affiche ensuite le flot de caractères émit par le client qui s'y connecte. Pour cela, le serveur reçoit en argument sur la ligne de commande le port sur lequel il va attendre les connexions des clients

A l'aide de Firefox, connectez-vous sur ce serveur et observez les informations transmises par Netscape. En déduire le codage d'une requête http (page voulue) ?

### ***Exercice 3***

Ecrire un client qui reçoit en argument :

- Le nom du serveur auquel on va chercher à se connecter
- Le numéro de port de ce serveur
- Un nom de fichier

Ce client doit se connecter à un serveur web standard (par exemple `intra.esiee.fr`) pour lui envoyer une requête http demandant l'affichage du nom de fichier donné en argument (par exemple `index.html`). Pour construire la requête, vous utiliserez les informations obtenues par le serveur de l'exercice 1.

Après avoir envoyé sa requête, votre client doit attendre et afficher les données envoyées par le serveur.

Vous vous limiterez au format de pages html pour ce TP

### ***Exercice 4***

Ecrire un petit serveur web qui réponde aux requêtes du client écrit précédemment. Ce serveur stockera dans un fichier local (`log_file`) toutes les requêtes servies ou non. Pour chaque requête ce fichier contiendra l'adresse de la machine du client, la date et le fichier demandé.

Ce serveur recevra 2 numéros de port en argument. Il répondra normalement aux requêtes http sur le premier port, et retournera son fichier de log si un client s'y connecte par le second port.