Université Paris Diderot – Paris 7 **Programmation réseau**
L3 Informatique **Année 2012-2013, 2ème semestre**

TP n°6

Programmation réseau en C

En C, tout est de bas-niveau, c'est à dire que contrairement au java beacoup de choses sont à faire "à la main", et il sera important d'avoir l'API C bien en tête, si nécessaire, une documentation est disponible sur la page du cours.
Voici un petit rappel.
Les prototypes des fonctions supplémentaires nécessaires se trouvent dans les fichiers en-têtes suivants.

```
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <arpa/inet.h>
```

Les fonctions principales sont :

```
int socket(int domaine, int type, int protocole);
int bind(int socket, const struct sockaddr *adresse, socklen_t longueur);
int listen(int socket, int attente);
int accept(int socket, struct sockaddr *adresse, socklen_t *longueur);
int connect(int socket, const struct sockaddr *adresse, socklen_t longueur);
int shutdown(int socket, int how);
int closesocket(int socket);
ssize_t send(int socket,const void *tampon, size_t longueur,int options); ou sendto
ssize_t recv(int socket,void *tampon, size_t longueur,int options); ou recvfrom
struct hostent *gethostbyname(const char *name); int h_errno;
int getpeername(int socket, struct sockaddr *name, int *namelen);
```

1 Clients en C

Exercice 1 [Un client TCP pour daytime]
Écrivez un programme C qui se connecte en TCP au service `daytime` sur `lucien` en utilisant des sockets.

Exercice 2 [Un client UDP pour daytime]
Refaites l'exercice précédent en utilisant cette fois des sockets UDP.

2 Serveur en C

Exercice 3 [Un serveur pour daytime]
Faites un serveur qui simule le service daytime.
Remarque : Dans une prochain TD vous verrez comment faire un serveur parallélisé en C à l'aide des processus (`fork`). Ici le serveur fermera la connection à chaque réponse.