

Objectifs du TD Informatisé :

Comprendre la correspondance entre l'adresse IP et le nom d'une machine.
Savoir obtenir une adresse et un nom d'une machine avec des sockets.

1 TP : Internet : Correspondance entre noms et adresses

La structure `addrinfo` utilisée par `getaddrinfo()` contient les membres suivants pour identifier un hôte :

```
struct addrinfo {
    int ai_flags;           /* AI_PASSIVE, AI_CANONNAME, ... */
    int ai_family;         /* PF_xxx */
    int ai_socktype;        /* SOCK_xxx */
    int ai_protocol;       /* 0 ou IPPROTO_xxx pour IPv4 et IPv6 */
    size_t ai_addrlen;     /* la taille de l'adresse binaire ai_addr */
    char *ai_canonname;    /* le nom complètement qualifié */
    struct sockaddr *ai_addr; /* l'adresse binaire */
    struct addrinfo *ai_next; /* la structure suivante dans la liste chaînée */
};

cf http://manpagesfr.free.fr/man/man3/getaddrinfo.3.html;
```

Elle est décrite dans `/usr/include/netdb.h`. Un ensemble d'appels divers existe pour récupérer une structure de ce type concernant un hôte.

Voici le début du manuel pour un de ces appel :

NAME
`getaddrinfo`, `freeaddrinfo`, `gai_strerror` - Traduction d'adresses et de services réseau

SYNOPSIS
`#include <sys/types.h>`
`#include <sys/socket.h>`
`#include <netdb.h>`

```
int getaddrinfo(const char *node, const char *service,
               const struct addrinfo *hints,
               struct addrinfo **res);
```

```
void freeaddrinfo(struct addrinfo *res);
```

```
const char *gai_strerror(int errcode);
```

L'argument `hints` pointe sur une structure `addrinfo` qui spécifie les critères de sélection des structures d'adresses de sockets renvoyée dans la liste pointé par `res`.

Exercice 1

1. Lire (et comprendre) les manuels sur `netdb.h`, ainsi que `getaddrinfo` et `getnameinfo`, `freeaddrinfo` et `inet_n_top` et comprendre les structures de ces fonctions.
2. A l'aide de `getaddrinfo()`, écrire un programme qui permet d'obtenir le numéro *IP* d'une machine en fonction de son nom. Le numéro sera alors affiché à l'écran.
3. En utilisant le programme précédent, écrire un second programme qui permet d'avoir l'affichage du nom officiel ainsi que la liste des alias de l'hôte. Il faudra alors utiliser `getnameinfo()`
4. Ecrire un troisième programme qui permet de passer le nom en paramètre dans la ligne de commande, s'il est saisi dans le programme.
5. Essayer les programmes avec des noms de machine en local sur le réseaux de l'université, puis avec des noms correspondants à des sites distants, que vous connaissez dans le monde. *Internet*.
6. Regarder le manuel pour la commande `getent`. Utiliser cette commande et comparer les résultats avec les réponses obtenues précédemment sur votre réseau local.