

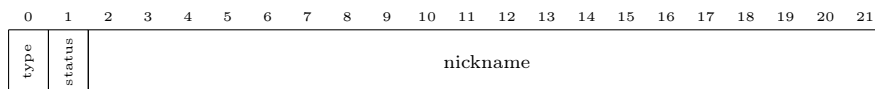
TP/TD 10 : Communication à travers le réseau

On souhaite réaliser un client de chat *peer to peer*.

Pour rendre les clients inter-opérables, nous nous fixons un protocole très simpliste, qui se base sur uniquement 2 types de packets :

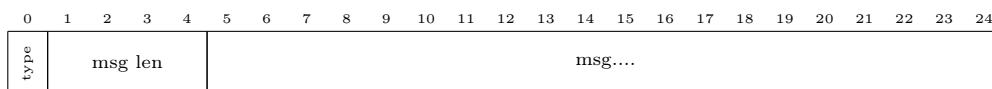
- type 1: packets permettant l'échange de noms et négociation du lien.
- type 2: packets transportant les messages des utilisateurs.

Le client désirant de communiquer avec le peer distant, devra envoyer un message de type 1, contenant son *nickname*. Le peer distant devra répondre avec le même type de packet, contenant le sien.



type	entier	1 octet	doit être 1 pour ce type de message
status	entier	1 octet	statut : 1 = succès. Tout autre numéro = échec.
nickname	tableau d'octets	20 octets	chaîne de caractères de exactement 20 octets. Si la chaîne initiale est plus petite, les octets restant doivent être remplis avec des 0

Une fois cet échange fait, le flux de communication doit être transféré uniquement avec des messages de type 2 :



type	entier	1 octet	doit être 2 pour ce type de message
msg len	entier	4 octet	taille du message qui suit
msg	tableau d'octets	taille variable	une suite d'octets de taille "msg len" contenant le message

Question 1

Vous aurez besoin des fonctions suivantes pour vous connecter à un serveur distant.

- *getaddrinfo*
- *getnameinfo*
- *socket*
- *connect*

A quoi sert chacune de ces fonctions?

Question 2

Écrivez le code du client qui

- se connecte au serveur;
- lui envoie le *nickname*;
- récupère le sien;
- effectue une communication en boucle

Note : Récupérez le binaire du serveur qui vous permettra de tester votre client et testez-le.

Question 3

Écrivez le code du serveur.