

Programmation réseaux

TD 6 programmation TCP en C

Mars 2007

Le but de ce TP est de voir comment implanter un client et un serveur TCP en C, en programmant un jeu de Blackjack simplifié.

1 Les règles du jeu d'un Blackjack simplifié

(d'après Wikipédia)

La partie oppose les joueurs à la banque, représentée par le croupier. Le but est de tirer des cartes pour approcher ou faire 21 points sans les dépasser. Dès qu'un joueur dépasse 21, il a perdu : on dit qu'il "saute" ou qu'il "crève". Pour simplifier, nous utiliserons ici seulement des cartes valant de 1 à 10 points, supposées équiprobables¹.

Le jeu se déroule en deux phases. Tout d'abord, tout joueur encore en dessous des 21 points peut demander une carte additionnelle au croupier en annonçant "Carte!". Il peut également à tout moment décider d'en rester là en déclarant "Servi!". C'est cette phase que nous allons simuler par un protocole client-serveur. Ensuite, une fois tous les clients servis, le croupier fait jouer la banque selon une règle mécanique : il tire des cartes jusqu'à atteindre ou dépasser 17. S'il dépasse 21, la banque saute et tout les joueurs encore en course gagnent. Si la banque est entre 17 et 21, la partie est gagnée, nulle, ou perdue selon que le joueur a plus, autant ou moins que la banque.

2 Le protocole

Le serveur va représenter le croupier, tandis que le client est un joueur. Les échanges seront constitués de messages d'une ligne (finie par `\n`). Le serveur peut émettre trois types de messages :

CUMUL : *n* Ce message signale que le joueur est encore en course. L'argument *n* est un entier entre 0 et 21 signalant le total actuel de points.

RESULTAT : *X msg* Ce message marque la fin du jeu. Le caractère *X* peut être **G**, **N** ou **P** pour signaler un gain, un nul ou une perte pour le joueur. Le *msg* qui suit est un petit texte explicatif, tel que par exemple **la banque atteint 22 et saute**. Ce texte peut être laissé vide dans un premier temps. Après l'envoi de ce message, le serveur doit clore la connection.

ERREUR : *msg* Le serveur signale qu'il a reçu un message incorrect. On annule le jeu et on termine la connection.

De son côté, le client va simplement répondre à chaque message **CUMUL** par un message **CARTE** ou un message **SERVI**.

¹s'il vous reste du temps, vous pourrez ajouter les cartes Valet, Dame et Roi, valant chacune 10 points, et permettre à l'As de valoir 1 ou 11 points, selon ce qui arrange le joueur.

Exercice 1 – *Un client de Blackjack*

Programmez en C un client implémentant le protocole ci-dessus : il devra pouvoir se connecter à un serveur de Blackjack, afficher le total de points actuel, permettre la saisie des décisions du joueur et l'envoi des messages correspondants au serveur, ainsi que l'affichage du résultat final.

Vous pouvez tester votre client avec le serveur fourni : `~letouzey/blackjack-serv`, voir l'option `--help` pour les indications d'usage.

Exercice 2 – *Un serveur de Blackjack*

Programmez en C un serveur de Blackjack tel que celui fourni. Testez-le à l'aide de votre client, ou d'une simple connection telnet.