

Communiquer et jouer en réseau

Aurèle Barrière & Rémi Hutin

5 avril 2016

Table des matières

1	Retransmission des matchs en directs	1
2	Poste mono-client	2
3	Compétition équitable	2
4	Bonus	2
5	Protocole	2

1 Retransmission des matchs en directs

2.1

Nous avons utilisé le code issu de nos deux projets. Les fonctions de mise à jour du plateau et de calcul de score viennent d'un des projets, tandis que l'affichage côté client vient de l'autre.

Le premier a l'avantage de ne considérer les couleurs que comme de caractères, qui s'envoient donc facilement par les *sockets* TCP.

Le second permet d'avoir, grâce à la SDL2 (www.libsdl.org/index.php), un affichage élaboré pour le client (qui permet aussi de recevoir les choix du joueur en cliquant simplement sur la couleur souhaitée).

2.2

On modifie ainsi le serveur. On commence par attendre une connection en acceptant sur un port (par exemple 7777).

Ensuite, on garde l'identifiant de la *socket* créée après acceptation et on lance une partie.

Au début, on envoie le plateau entier sur cette *socket*, puis on envoie le coup joué et le numéro du joueur à chaque tour.

Quand la partie est finie, on envoie à la place du prochain coup un signal de fin de partie.

2.3

Ainsi, il suffit de créer un client qui suive le schéma suivant :

- On se connecte sur le port demandé (par exemple 7777, qui est non réservé).
- Tant qu'on a pas reçu un signal de fin de partie (par exemple, le premier caractère du buffer est le caractère spécial '*'), on reçoit sur la *socket* utilisée pour la connection soit le plateau entier, soit le dernier coup joué et le numéro du joueur.
- Grâce aux fonctions de mise à jour et d'affichage, on retransmet la partie en cours.

2.4 (Bonus)

2.5 (Bonus)

2.6

2 Poste mono-client

2.7

2.8

2.9

2.10

2.11

3 Compétition équitable

2.12

2.13

4 Bonus

5 Protocole