

CASANOVA--FRANGER Alexandre
RAHARISON Ny Andry

Rapport de Projet Réseau

2016-2017

Après avoir analysé le sujet et compris le fonctionnement du morpion, nous avons passé du temps à comprendre la fonction main qui nous permet de jouer contre un ordinateur. Nous nous sommes intéressés dans un premier temps à gérer la mise en réseau du jeu via des sockets, nous définissons une fonction serveur et une fonction client au sein du même programme de manière à ce que si on le lance sans argument, on lance le serveur, et avec argument nous sommes le client et l'argument correspond à l'hôte. Tout d'abord nous faisons seulement des tests et nous ne intéressions pas encore au design de notre jeu. Ensuite il faut évidemment modifier le main pour que le jeu soit entre deux joueurs sur deux machines et non un joueur contre une machine.

Ceci fut notre première version, puis nous avons implémenté une version plus complète avec envoi de messages. Il a fallut créer un protocole au moyen des fonctions :

```
def envoi_message(sock, data)
def reception_message(sock)
def recvall(sock, count)
def envoi_int(sock, data)
def reception_int(sock)
```

Nous avons également défini des fonctions d'affichage pour interagir entre les joueurs connectés qui sont

```
def message_pour_tous(message) qui envoie un  
message serveur vers tous les clients,  
def redirection_affichage(grid, current_player), qui  
redirige l'affichage, et enfin  
def affichage_observateur (grid, current_player, shot,  
message) qui envoie des informations aux observateurs  
sur la partie en cours.
```

Maintenant que notre programme fonctionne nous avons désigné un peu notre morpion au moyen de "clear" sur le terminal et nous avons géré les répétitions pénibles de messages en cas de placement de pièce sur une pièce adverse. Nous avons implémenté la possibilité de pouvoir rejouer une partie à la fin d'une partie en choisissant de répondre Y ou N (yes or no) pour la question "Voulez-vous rejouer une partie ?" On a la possibilité que des joueurs puissent devenir des observateurs car l'on a créé un dictionnaire regroupant les observateurs mais le problème est que l'on arrive pas à transmettre des messages aux observateurs. Nous n'avons pas réussi à gérer le fait qu'un client puisse quitter la partie via un control-C inattendu et les problèmes de communications.

Ce qui nous a le plus posé problème durant ce projet était l'implémentation du protocole et les problèmes d'affichage de grille. Nous nous excusons pour le github, le PC de Ny Andry est tombé en panne et nous

avons dû nous rassembler tous les deux en ne travaillant que sur mon PC uniquement, cela explique que les commit n'ont été effectués que par moi seulement et très tardivement car nous n'avions pas le besoin de commit cela car nous travaillions directement via mon PC. Merci de votre compréhension