

# Le morpion aveugle - mise en réseau

Vous avez déjà jusqu'ici participé à la réalisation d'un chat en réseau, faisant usage d'un protocole relativement simple. Maintenant, passons à l'étape supérieure.

## Protocole

Le protocole que nous allons utiliser est relativement simple. Il va permettre de couvrir les fonctionnalités suivantes :

- Connexion et déconnexion de joueurs
- Possibilité de jouer une ou plusieurs partie
- Possibilité de regarder une partie
- Reconnexion d'un joueur après une déconnexion "propre"
- Communication des scores
- Gérer la fin de partie

Mot-clé	Source	Arguments	Description
JOIN	C	-	Permet de rejoindre la partie.
END	S	Joueur perdant	Indique au client cible la fin de partie.
DISCONNECT	C	-	Se déconnecter proprement.
STATE	S	Grille de jeu	Communique l'état d'une grille sous la forme de 9 entiers concaténés séparés par un espace.
START	S	Premier joueur	Indique au client le début de partie et le joueur qui commence.
YOURTURN	S	-	Indique au client qu'il doit jouer.
END	S	Gagnant	Indique au client la fin de partie.
SCORE	S	Scores	Communique les scores au client sous la forme de 2 entiers concaténés séparés par un espace, ordre J1 > J2
PLACE	C	Case	Communique au serveur l'intention d'un client de jouer à la case spécifiée.

Note : la réception d'un message vide depuis le client de la part du serveur entraîne la déconnexion du client. Permet de gérer certains crashes serveur

## Cas typique

Typiquement, une partie se déroulera de cette façon là :

<ul style="list-style-type: none"><li>• Connexion de clients au serveur</li></ul>
<ul style="list-style-type: none"><li>• Début de partie quand assez de joueurs - envoi de START</li></ul>
<ul style="list-style-type: none"><li>• Envoi de STATE x à tous les clients et YOURTURN au joueur courant</li></ul>
<ul style="list-style-type: none"><li>• Envoi de PLACE x depuis le joueur courant au serveur</li></ul>
<ul style="list-style-type: none"><li>• Répétition des deux dernières étapes jusqu'à un état de fin de partie</li><li>• Envoi de END x et SCORE y à tous les clients</li></ul>
<ul style="list-style-type: none"><li>• Attente du JOIN de deux clients pour une nouvelle partie</li></ul>

A la fin du stage, vous aurez géré ceci et ajouté vos personnalisations.

## Utile

Les fichiers et leurs rôles :

- server.py : toute la logique du programme hôte. C'est ici que vous gèrerez la réception de flux entrants du serveur.
- client.py : toute la logique du programme client. C'est ici que vous gèrerez la réception de flux entrants du client.
- game.py : toute la logique d'une partie. C'est ici que vous gèrerez les communications liées à la partie (envoi d'états aux clients etc).
- outils.py : quelques fonctions utiles
- cmdServer.py : quelques outils pour le serveur
- cmdClient.py : quelques outils pour le client
- grid.py : logique de la grille. Pas besoin d'y toucher

**Le code contient plein de fonctions utiles dans presque tous les fichiers, prenez bien soin de les lire en entier attentivement et essayez de réutiliser ces fonctions au maximum. Cela réduira le travail nécessaire pour certaines fonctionnalités et vous forcera à comprendre le code fourni dans son ensemble..!**

## A faire :

- Compléter les fichiers pour gérer le cas typique d'une partie montré plus haut. Il suffit de compléter les "TODO"
- Gestion des spectateurs, de la logique y est mais ils ne sont pas gérés complètement
- Amélioration du système de score, j'en parlerai plus en détail à l'oral.
- Améliorations personnelles. Si vous n'avez pas d'idée, je peux toujours vous en trouver..