

PROJET DE RESEAU



Licence 3

LE JEU DU MORPION AVEUGLE

Projet de réseau de semestre 5 du parcours Licence Informatique. Mise en réseau d'un jeu de morpion aveugle.

Projet réalisé par Benjamin de POURQUERY et Quentin LEMONNIER, étudiants à l'Université de Bordeaux, collège sciences et technologies.

PROJET DE RESEAU

LE JEU DU MORPION AVEUGLE

Table des matières

PROTOCOLE UTILISE.....	2
INTRODUCTION	2
COMMANDES.....	2
MISE EN SITUATION	3
PERSPECTIVES D'AMELIORATION	4
GESTION D'UN STATUT DE PARTIE	4
RENDRE LE NOMBRE DE JOUEURS MALLEABLE	4
GESTION DE L'ABSENCE D'UN JOUEUR.....	4
AJOUT D'UN ROBOT JOUEUR	4
BILAN.....	5
REALISE.....	5
A AMELIORER.....	5
PROBLEMES RENCONTRES.....	5
REMERCIEMENTS	5

PROTOCOLE UTILISE

INTRODUCTION

Ci-dessous sont listées les commandes échangées entre le serveur et les clients. Il s'agit du protocole que nous avons établi pour que serveur et clients puissent communiquer correctement. Cela implique :

- Pouvoir envoyer les commandes nécessaires à l'avancement du jeu du côté des clients
- Pouvoir communiquer l'état de jeu propre à un client du côté du serveur
- Effectuer la gestion de partie dans son ensemble

COMMANDES

STATE :

Source : Serveur

Arguments : grille de jeu sous la forme de 9 entiers concaténés, séparés par un espace.

Description :

Commande envoyée depuis le serveur aux clients, communiquant l'état de jeu approprié. Le client transforme la chaîne en grille afin de la rendre plus lisible.

START :

Source : Serveur

Arguments : Numéro du joueur

Description :

Commande indiquant aux clients que les deux joueurs sont connectés et que la partie va débiter.

SCORE :

Source : Serveur

Arguments : Scores des deux joueurs concaténés, séparés par un espace. Ordre : J1 > J2

Description :

Commande indiquant aux clients les scores des joueurs. Mise en forme côté client pour plus de lisibilité.

YOURTURN :

Source : Serveur

Arguments : Aucun

Description :

Commande « token » indiquant à un joueur qu'il doit jouer.

END :

Source : Serveur

Arguments : Joueur ayant gagné la partie

Description :

Commande indiquant au client la fin de partie. Mise en forme côté client.

JOIN :

Source : Client

Arguments : Aucun

Description :

Commande permettant à un spectateur de rejoindre la partie. Utilisé en fin de partie pour permettre aux clients de jouer une nouvelle partie.

DISCONNECT :

Source : Client

Arguments : Aucun

Description :

Commande du client à utiliser pour se déconnecter « proprement ».

MISE EN SITUATION

Le serveur se lance. Il va attendre les clients.

Les clients se connectent, le serveur les affecte à la partie gérée par une classe « Game ». Ils reçoivent l'état actuel de la grille.

Une fois que le serveur dispose de ses deux joueurs, il envoie « START N°Joueur » aux joueurs, leur indiquant leur numéro. Le serveur envoie également au premier joueur « YOURTURN », lui signifiant qu'il doit jouer.

Le joueur courant saisit une case où il souhaite jouer. Son client va envoyer au serveur « PLACE N°Case ».

Le serveur traite de façon appropriée la réception du « PLACE » :

- Si le joueur joue sur une case correcte (i.e. vide pour les deux joueurs), les grilles de jeux sont correctement mises à jour. Le serveur renvoie le nouvel état de grille au joueur qui vient de jouer et indique à l'autre joueur qu'il doit jouer.
- Si le joueur joue sur une case précédemment jouée par son adversaire mais qu'il n'en a pas connaissance, le serveur met à jour sa grille et lui renvoie, lui indiquant qu'il doit jouer à nouveau.
- Si le joueur joue sur une case qu'il a déjà jouée, le serveur lui demande de rejouer, en renvoyant également l'état de la grille.

À la victoire d'un joueur, le serveur envoie « END N°Joueur gagnant » aux deux joueurs. Il envoie ensuite « SCORE ScoreJ1 ScoreJ2 » aux clients, le score ayant été mis à jour au préalable. Il replace les joueurs en spectateurs et leur indique qu'il faut taper la commande « JOIN » pour pouvoir rejouer.

Le serveur attend les « JOIN » des spectateurs pour lancer une nouvelle partie.

PERSPECTIVES D'AMÉLIORATION

GESTION D'UN STATUT DE PARTIE

Utiliser une énumération de statuts. Permettrait une amélioration de la gestion de la partie : pause, en cours, attente de joueurs, ... Actuellement le jeu dispose seulement d'un booléen « gameReady » qui indique si le jeu est en cours ou non. Cela pourrait par exemple la mise en place d'un « lobby » d'attente pour les joueurs.

RENDRE LE NOMBRE DE JOUEURS MALLEABLE

Couplé à la gestion du statut de partie, avec la mise en place d'un lobby, ajouter des commandes de configuration du nombre de joueurs pourrait permettre de faire des parties à plus de deux joueurs, tout en modifiant les fonctions d'ajout de joueurs de façon appropriée.

GESTION DE L'ABSENCE D'UN JOUEUR

Possibilité d'ajout de « timers » côté serveur pour chaque joueur, lancé à l'envoi de YOURTURN. Permettrait ainsi de gérer l'absence d'un joueur et de finir la partie dans ce cas.

AJOUT D'UN ROBOT JOUEUR

Possibilité d'ajout d'un robot qui joue à la place d'un joueur. Couplé à la gestion du statut de partie, la réalisation d'une IA qui joue aléatoirement serait possible.

BILAN

REALISE

Mise en réseau du jeu de morpion aveugle

Mise en place de spectateurs

Gestion de déconnexion

Gestion de la reconnexion d'un joueur

A AMELIORER

Toutes les perspectives précédemment énoncées peuvent valoir ici.

Par ailleurs, une modification de la gestion des clients permettrait de mieux gérer la reconnexion du joueur. En effet, il faudrait enregistrer d'une meilleure façon l'identité des clients, car si un joueur s'étant déconnecté involontairement souhaite se reconnecter et qu'un autre client se connecte entre temps, il prendra sa place au cours de la partie.

PROBLEMES RENCONTRES

Barrière du langage par moments, Python n'étant que peu connu de notre groupe.

Lors de la saisie d'une case par le client, nous n'avons pas réussi à gérer la perte de connexion au serveur. Il aurait peut-être fallu gérer l'entrée « `sys.stdin` » autrement et l'utiliser pour les « PLACE ». Nous n'avons pas eu le temps d'essayer cela.

Problème d'organisation, faute duquel nous aurions pu réaliser davantage de choses. Les perspectives d'amélioration énoncées précédemment paraissent à l'état de conception mais leur implémentation, bien que non effectuée, a été clairement envisagée du point de vue algorithmique.

REMERCIEMENTS

Nous souhaitons remercier nos professeurs de réseau, autant notre chargé de TD Monsieur DESPRET Vincent que Monsieur THIBAUT Samuel, pour l'enseignement précieux qu'ils nous ont apporté.

Auteurs : Benjamin de POURQUERY – Quentin LEMONNIER