21/04/2016

Contaret Allan

Ivan PERIC

Morpion

En mode console



**Morpion en mode console**

**Introduction**

Le but du projet est créer un morpion dans le langage choisi (ici le C++) qui puisse être

joué depuis la console.

**Fonctionnalités**

Le jeu devra donc prendre en compte le quadrillage, les deux joueurs, les signes X et O

ainsi que la vérification de l’état du plateau pour savoir quel joueur a gagné. Une interface

devra exister soit un menu qui permettra au joueur de choisir entre plusieurs options qui

sont les règles du jeu, jouer au jeu ou alors quitter.

Pour ce faire, il faudra créer plusieurs classes dans le code pour qu’on puisse séparer le

Code du joueur et celui du plateau.

**Répartition des tâches**

|  |  |
| --- | --- |
| Allan | Ivan |
| * Création de la classe joueur avec l’attribution des signes X et 0 pour chaque joueur * Création des tours grâce à une boucle pour permettre à chaque joueur de jouer l’un après l’autre * Création de conditions permettant de vérifier si une case est remplie ou non pour y insérer le signe du joueur | * Création du Plateau avec l’apparence de celui-ci * Attribution de chiffres pour chaque case en les insérant dans un tableau pour pouvoir y afficher les X et les O * Création du menu avec les choix entre règles, jeu et quitter le jeu * Création de la documentation |

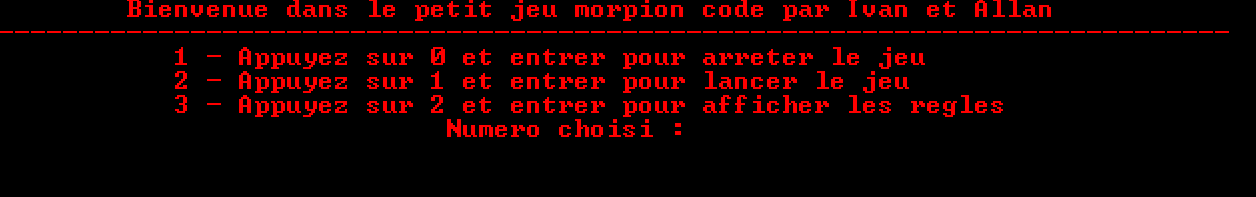
**UML**

|  |
| --- |
| Joueur |
| + tableau[3][3] : char  + tour : char  - choix : int  - ligne : int  - colonne : int |
| + Joueur()  + ~Joueur()  + tour\_joueur : void  - tab\_choix() : char |

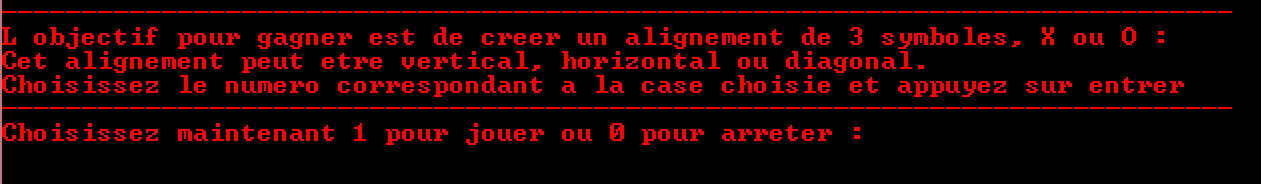
|  |
| --- |
| Plateau |
|  |
| + Plateau()  +~Plateau()  + lancement\_plateau (Joueur joueur) |

**Le déroulement du programme**

Tout d’abord le joueur arrive sur le menu :



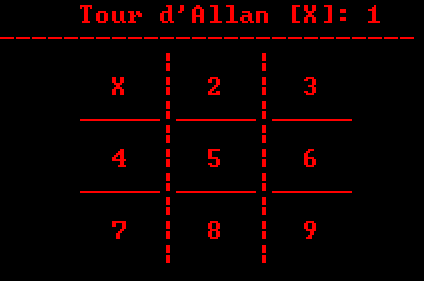
Ensuite soit il choisit d’afficher les règles :



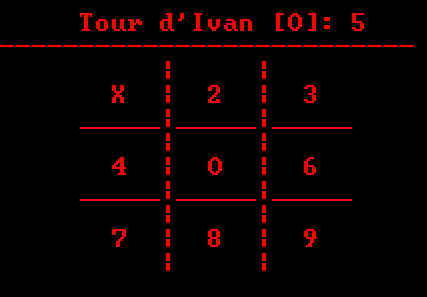
Ou bien de choisir de jouer :



Si il décide de jouer il doit placer son pion. Par exemple le X sur la case 1 :



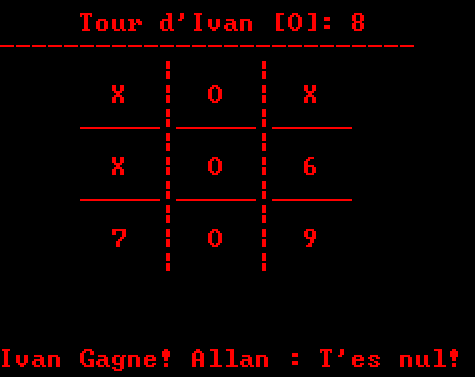
Du coup c’est au tour de l’adversaire :



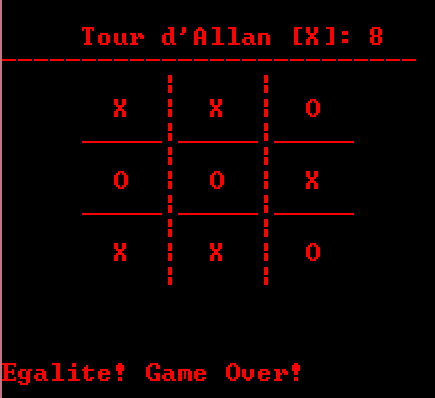
Si le joueur 1 (X) gagne :



Si le joueur 2 (O) gagne :



Et si il y a égalité :



**Ce que nous a appris le sujet**

Ce projet nous a appris à mieux utiliser les classes et les interactions entre elles. En ayant eu des difficultés sur ce point, nous avons pu nous améliorer pour mieux réaliser notre projet. Nous avons aussi appris à mieux utiliser Git hub vu que l’on ne l’utilise pas souvent.

**Problèmes rencontrés**

Nous avons rencontrés deux problèmes majeurs lors de la réalisation du programme du

jeu. Le problème étant le fait que lorsque le premier joueur choisi sa case, le programme ne passait

pas au deuxième joueur et restait donc sur le premier. Le deuxième problème est que le programme

réalisait la vérification de la victoire ainsi que l’égalité mais n’affichait pas les X et O. Avec de la difficulté

et de la recherche, ces problèmes ont été réglés.

**Solutions trouvées**

Pour notre premier problème, c’est-à-dire le changement de joueur, nous avons fait passés la variable « tour » en variable globale.

Et en ce qui concerne notre deuxième problème, le souci de l’affichage des pions, nous avons supprimé les tableaux pour les avoir que dans la classe « Joueur » et nous avons placé le joueur en paramètre dans la fonction de lancement de plateau.