
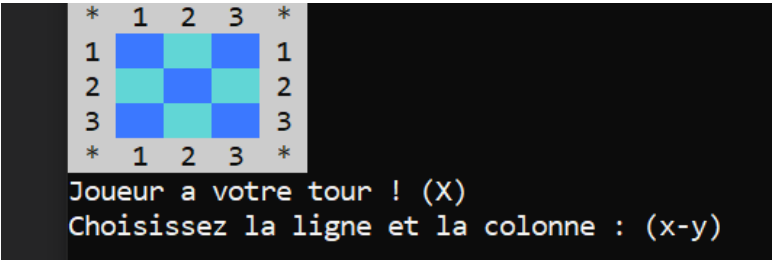


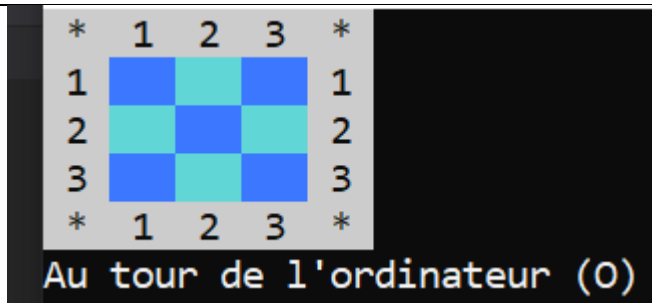
DESCRIPTION D'UNE MISSION BTS SIO			
Prénom – Nom	Quentin Coqueran	N° mission	2
Option	SLAM		
Situation		Formation	
Lieu de réalisation	Campus Montsouris 2 Rue Lacaze – 75014 PARIS		
Période de réalisation	Novembre 2019	Décembre 2019	
Modalité de réalisation	VÉCUE		

Intitulé de la mission	Titre de la mission
	Création d'un morpion C#
Description du contexte de la mission	Description en 2 à 3 lignes maxi
	Créer un morpion sur l'invite de commande en c# contre L'IA avec plusieurs modes de difficulté.

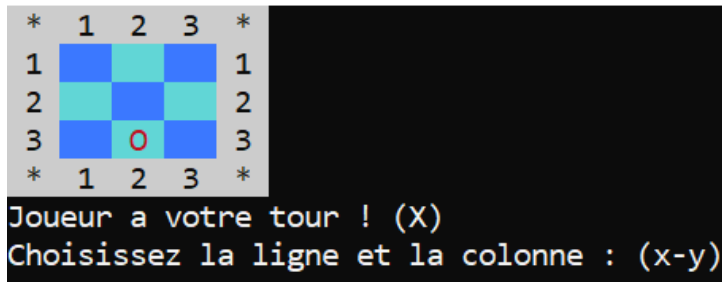
Ressources et Outils utilisés	Liste des ressources disponibles et outils utilisés (Documentations, Matériels et Logiciels)
	- Visual Studio 2019
Résultat attendu	Résultat attendu avec la réalisation de cette mission
	Que le joueur puisse jouer sans difficultés contre l'IA avec plusieurs modes de difficultés selon ses envies.
Contraintes	Contraintes : techniques budgétaires temps O.S. ou outils imposés...
	<ul style="list-style-type: none"> - Faire en sorte que l'interface graphique soit intuitive pour l'utilisateur - Créer plusieurs modes de difficultés différentes et permettre à l'utilisateur de choisir son mode. - Gérer les exceptions du joueur ou bien ses erreurs

Compétences associées (voir tableau) à faire	Liste des intitulés du tableau de compétences (avec les références)
	<p>A1.1.3 : Étude des exigences liées à la qualité attendue d'un service</p> <p>A1.3.3 : Accompagnement de la mise en place d'un nouveau service</p> <p>A2.1.1 : Accompagnement des utilisateurs dans la prise en main d'un service</p> <p>A2.1.2 : Évaluation et maintien de la qualité d'un service</p> <p>A2.1.2 : Évaluation et maintien de la qualité d'un service</p> <p>A5.1.1 : Mise en place d'une gestion de configuration</p> <p>Réalisation des tests nécessaires à la validation d'éléments adaptés ou développés</p>

Description simplifiée des différentes étapes de réalisation de la mission en mettant en évidence la démarche suivie, les méthodes et les techniques utilisées
<p><u>Chapitre 1 : Création du projet et du tableau de jeu</u></p> <p><u>Etape :</u></p> <p>1.1/ Tout d'abord nous créons un projet Application console.</p> <p>1.2/ Ensuite on crée une procédure Affichage qui va créer un tableau 5x5 qui nous permettra de faire le plateau pour tous les modes de difficulté.</p>  <p><u>Résultat :</u></p> <p>Le plateau de jeu est prêt</p> <p><u>Chapitre 2 : Création des différentes difficultés</u></p> <p><u>Etape :</u></p> <p>2.1/ Nous allons créer 3 différents modes de difficultés, le mode normal où l'IA joue aléatoirement sur les cases non remplies, le mode défensif où l'IA fera tout pour vous empêcher de gagner et le mode offensif où si l'IA a le choix entre gagner et vous empêcher de gagner il choisira de gagner.</p> <p>2.2/ Pour l'IA normal on crée une boucle qui génère une position aléatoire et qui vérifie si la case est remplie. Si la case est vide l'IA remplit la case avec le signe de l'IA.</p> <p>Avant avoir jouer</p>

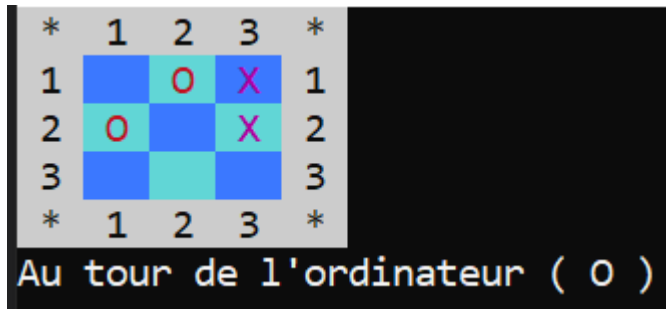


Après avoir jouer

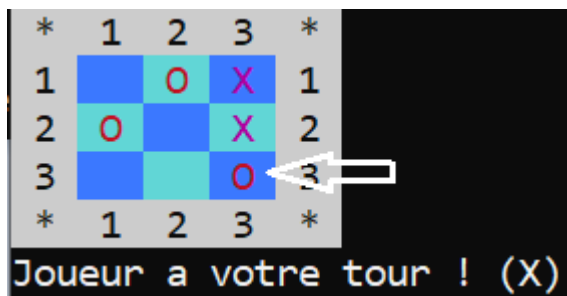


2.3/ Pour l'IA défensif on utilise l'IA normal sauf qu'on rajoute une condition qui vérifie si l'adversaire est sur le point de gagner et le bloque.

Avant avoir jouer

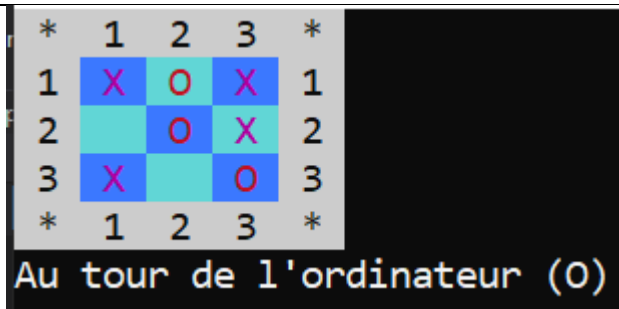


Après avoir jouer

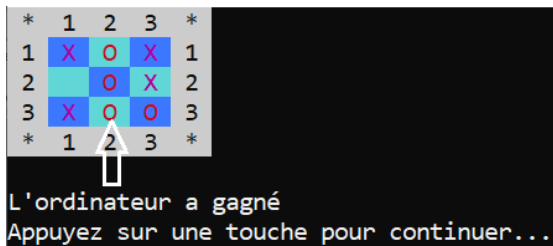


2.4/ Pour l'IA offensif on utilise l'IA défensif sauf qu'on rajoute une condition qui vérifie si l'IA peut gagner plutôt que de bloquer l'adversaire.

Avant avoir jouer



Après avoir jouer



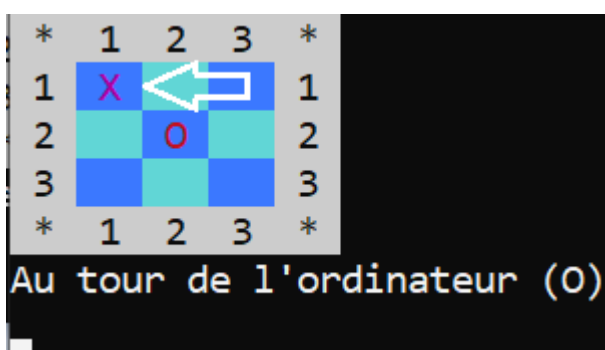
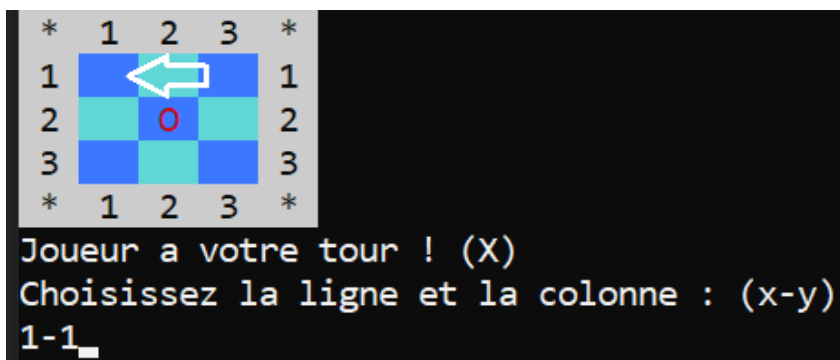
Résultat :

L'IA a désormais différentes difficultés

Chapitre 3 : Création joueur

Etape :

3.1/ Nous allons créer une procédure qui va permettre au joueur d'entrer les coordonnées sur une seule ligne avec la fonction split puis gérer si la case est remplie ou non.



Résultat :

Le joueur peut jouer

Chapitre 4 : Création menu et fin de partie

Etape :

4.1/ On va créer le menu pour pouvoir choisir son niveau de difficulté. Le joueur peut entrer son choix en fonction de la difficulté.

```
+-----+
| Partie normale : 1 |
| Partie defensif : 2 |
| Partie offensif : 3 |
| Quitter : 4 |
| Votre choix : 2_ |
+-----+
```

4.2/ On créer une fonction qui va vérifier qui a gagné après chaque coup jouer ou si le tableau est rempli la fonction retourne match nul.

	Que pouvez-vous dire de cette mission : apport personnel, expérience, etc
Conclusion	Meilleure compréhension des boucles, des exceptions et des fonctions et procédures. Amélioration dans le langage C#.

Productions associées	Liste des documents produits et description