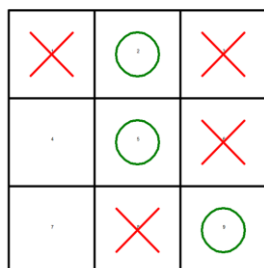


Un enseignement au numérique pour notre académie : Le morpion dominé

Le morpion est à l'origine un mot qui fait partie du domaine de la biologie et qui désigne un pou, petit animal qui a pour fâcheuse tendance de se "scotcher" sur la peau afin d'y déposer ses larves.

Source : site Wikipédia
Domaine public



Le morpion désigne aussi un jeu bien connu des élèves, jeu qui a la fâcheuse tendance à s'inviter discrètement pendant les heures de cours et à se scotcher sur le cahier des élèves, là où il n'a rien à y faire.

Notre intention n'est nullement ici de faire du prosélytisme pour ce jeu qui peut occuper passivement les élèves mais au contraire de montrer comment il peut devenir **un jeu intelligent**, c'est à dire un jeu permettant de développer des capacités à percevoir une situation et à concevoir une réponse au regard d'une stratégie bien étudiée.

Pour ce faire, nous allons découvrir dans un premier temps comment l'on peut arriver à décortiquer ce jeu en faisant ressortir le fait que les situations possibles sont parfaitement identifiables et construire ainsi une stratégie non-perdante. Nous vous proposerons ensuite de programmer de A à Z sous Python le jeu de morpion, vous pourrez ainsi tester ladite stratégie.

Une analyse du jeu de morpion

Yves Guichard

Institut de Recherche sur l'Enseignement des Mathématiques de Lyon

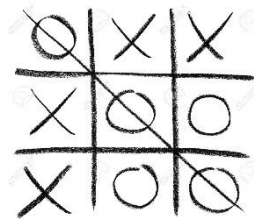
Historique

Le jeu du Tic Tac Toe est un jeu d'alignement à deux joueurs, une variante simplifiée du jeu du « Morpion » (jeu qui se joue sur une grille qui n'a pas de limites). On trouve une version de ce jeu dans l'empire romain, vers le premier siècle avant notre ère. Le nom Tic Tac Toe est un mot d'origine anglaise.

Matériel

Une feuille blanche et des crayons. On trace sur la feuille une grille carrée de 3 lignes et 3 colonnes.

L'objectif est d'aligner 3 symboles identiques.



Les deux joueurs doivent remplir, à tour de rôle, une case de la grille avec le symbole qui leur est attribué, généralement O et X. Celui qui aligne, le premier, 3 symboles identiques, a gagné. L'alignement peut être, horizontal, vertical ou en diagonale. L'issue du jeu peut être la victoire de l'un des deux joueurs ou une partie nulle.

Comme la grille comporte 9 cases, le nombre de combinaisons est limité. Il y a 9 possibilités pour le premier coup, 8 pour le second ce qui donne 72 pour les deux premiers. Donc en tout il y a $9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1$ possibilités de coups à ce jeu soit 362880 parties distinctes. Par ailleurs, on peut remarquer d'abord qu'en raison des symétries de la grille, il y a 3 positions initiales possibles : au milieu de la grille, au sommet d'une diagonale, à l'un des sommets de la croix centrale. Il est donc simple d'analyser ce jeu. Le jeu de Tic Tac Toe est un jeu à information complète.

Il n'est pas question, d'examiner toutes les situations possibles mais à partir du moment où les deux premiers coups sont joués, les coups sont contraints et, si les deux joueurs jouent bien, la partie s'achève sur un nul. X est le premier joueur. Il a le choix entre trois positions: le centre de la grille, un des coins de la grille ou une des cases de la croix.

Dans cette partie le joueur X commence au centre.

Si par exemple le joueur O joue sur la croix, on voit que cette réponse conduit à une défaite si le joueur X joue bien (voir les grilles ci-dessous). En effet, le joueur X (grille 3) impose le deuxième coup de O qui va empêcher la diagonale (grille 4), et en mettant sa croix sur la case appropriée (grille 5), deux alignements sont commencés et le joueur O ne peut en contrer qu'un seul.

										O				O				O						X
	X			X				X			X				X	X		O	X	X		O	X	X
				O				O	X		O	X			O	X			O	X		O	O	X
1				2				3			4			5				6				7		

En revanche, en répondant sur un coin, le joueur O force cette fois-ci la partie nulle.

							X				X				X			X		O		X		O
	X				X			X			X			X			X			X			X	X
					O				O		O	O	X	O	O	X	O	O	X	O	O	X	O	O
1					2			3			4			5			6			7				

Partie 2

Dans cette partie le joueur X commence sur une case des diagonales.

	O				O	X			O	X			O	X			O	X		X	O	X		X	O	X		X	O	X
									O				O				O				O			O	O			O	O	X
X				X				X				X	X			X	X	O		X	X	O		X	X	O		X	X	O
1				2				3				4				5			6			7					8			

La partie est nulle.

		O				O				O				O		O		O		O
					X				X	O			X	O		X	O		X	O
X				X				X				X		X		X		X	X	X
1				2				3				4				5			6	

Le joueur X gagne la partie.

									X				X			X			O	X			O	X	X		
	O			O			O		O			O	O		X	O	O		X	O	O		X	O	O		
X			X	X	X	O	X		X	O	X		X	O	X	X	O	X		X	O	X		X	O	X	
1				2				3				4				5			6			7					8

La partie est nulle.

Partie 3

Dans cette partie le joueur X commence sur une case de la croix.

						X			X			X			X		O		X		O		X		O		
X			X			X			X			X			X			X	X			X	X	O			
				O			O		O		O	O	X	O	O		O	X	O		O	X	O		O	X	O
1				2				3				4				5			6			7					8

Le joueur O gagne la partie.

Conclusion

On observe que le joueur qui commence la partie, ne peut pas perdre s'il joue la case centrale en premier. Si le deuxième joueur joue ensuite dans une des cases de la croix, il perd. Sinon le joueur X peut neutraliser le jeu.

Pour gagner, un joueur doit marquer une case qui lui ouvre le plus de possibilités par la suite.

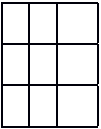
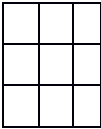
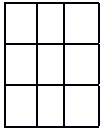
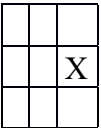
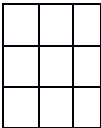
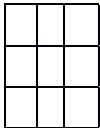
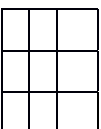
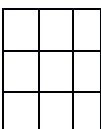
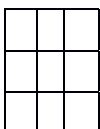
On peut noter que si les deux joueurs jouent bien, le jeu ne peut se terminer que par une partie nulle.

Variantes du jeu

1^{ère} variante : il est possible de jouer sur une grille 4x4 ; le but du jeu est d'aligner 4 symboles identiques.

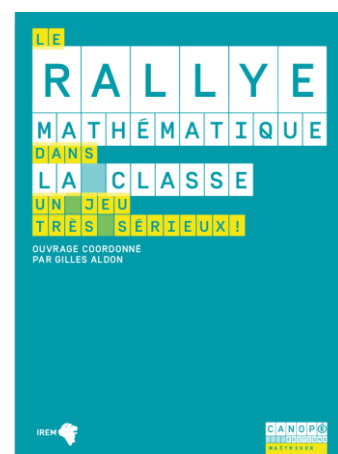
On peut jouer également sur un quadrillage "illimité" et varier le nombre de symboles à aligner.

2^{ème} variante : Ultimate Tic Tac Toe

1  2  3  4  5  6  7  8  9 	C'est un jeu à deux joueurs X et O. X joue le premier. Le jeu se joue sur une grille de 3x3 de 9 cases qui contiennent chacune une mini grille de 3x3. La grille comporte donc en tout 81 cases. Les règles sont celles du Tic Tac Toe complétées. Les joueurs remplissent tour à tour les cases avec le symbole qui leur est attribué. Lorsqu'un joueur a réussi à aligner 3 symboles dans une mini-grille il remporte cette mini grille. Le premier joueur qui a gagné trois mini-grilles alignées (horizontalement, verticalement ou en diagonale) remporte la partie. Ces règles sont complétées par les trois règles suivantes: - Si un joueur joue dans une case, le joueur suivant joue dans la mini-grille qui occupe la même place dans la grille que le coup joué. Ainsi dans l'exemple ci-dessus, si X joue dans la grille 4 comme indiquée, O devra jouer dans la grille 6. - Si un joueur envoie son adversaire dans une mini-grille déjà gagnée, l'adversaire devra jouer sur une case vide de cette mini-grille. Si celle-ci est pleine, l'adversaire pourra jouer dans n'importe quelle case vide de la grille de jeu.
--	--

Ce texte est extrait de l'ouvrage édité par Réseau Canopé en partenariat avec l'IREM de Lyon "Le rallye mathématique dans la classe - Un jeu très sérieux !"

© Réseau Canopé, 2018



Le jeu de morpion sous Python

Jean-Alain Roddier

IA-IPR de mathématiques

Programmons notre ordinateur en utilisant le logiciel Python. Ce logiciel fait l'unanimité pour ce qui est de rendre la programmation accessible à tous. Python est tellement diffusé que l'on parle aujourd'hui de Langage Python pour désigner la syntaxe et la forme des lignes de codes exécutables par ce logiciel.

Dans un premier temps, nous téléchargerons le logiciel Python puis nous l'utiliserons pour programmer notre jeu de morpion.

Téléchargement du logiciel Python

Dans la fenêtre de saisie de tout bon moteur de recherche saisissez : Télécharger Python. Vous allez ainsi avoir accès au lien suivant : <https://www.python.org/downloads/>, lien qui permet de télécharger la dernière version (en l'occurrence 3.7) du logiciel Python.



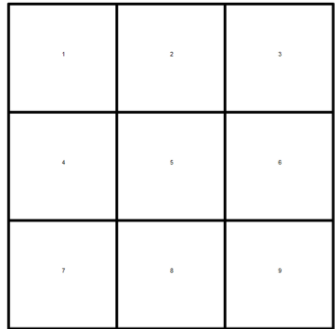
Programmation du jeu de morpion

Nous allons diviser cette partie de programmation en trois paragraphes : un paragraphe d'outils, un paragraphe de construction de la grille à neuf cases du jeu et un paragraphe qui donne les outils pour jouer.

Un paragraphe d'outils: ouverture d'une bibliothèque et définition de deux listes

Programmes	Explications
>>> from turtle import *	Cette instruction active une bibliothèque d'instructions de géométrie (aller à un point donné du plan, baisser ou lever le stylo, ...).
>>> List = [1,2,3,4,5,6,7,8,9]	On définit la liste List de 9 valeurs qui permettra d'identifier chacune des 9 cases de la grille.
>>> Point = [(-200,200),(0,200),(200,200),(-200,0),(0,0),(200,0),(-200,-200),(0,-200),(200,-200)]	On définit la liste Point des 9 coordonnées des points qui sont aux centres des cases.

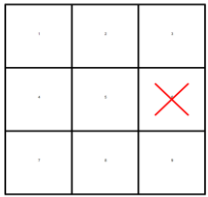
Un paragraphe pour construire la grille à neuf cases.

Programmes	Explications
<pre> >>> def grille(): reset() up() goto(-300,300) down() width(5) goto(-100,300) goto(-100,-300) goto(100,-300) goto(100,300) goto(300,300) goto(300,100) goto(-300,100) goto(-300,-100) goto(300,-100) goto(300,-300) goto(-300,-300) goto(-300,300) goto(300,300) goto(300,-100) i=0 while i<9: up() goto(Point[i]) down() write(List[i]) i=i+1 up() goto(-350,0) </pre>	Le programme s'appelle grille. On efface tout ce qu'il y avait sur l'écran. On lève le stylo afin qu'il ne marque rien Le pointeur va se placer sur le point de coordonnées (-300,300) puis on abaisse le stylo. On règle la largeur des traits qui vont être tracés, le stylo se rend ensuite au point de coordonnées (-100,300). Cela permet de tracer un des segments de la grille. On poursuit ainsi la construction de la grille jusqu'à l'obtenir complètement (vous constaterez que nous avons choisi de réaliser un joli serpent). On passe ensuite au marquage de chaque case à l'aide d'un numéro allant de 1 à 9. La liste Point est en effet parcourue de son rang numéro 0 au rang numéro 8, ce qui correspond à parcourir la grille d'un centre de case à un autre. On veille pour chaque point parcouru à marquer le numéro correspondant qui est donné par l'élément correspondant de la liste nommée List. A la fin, on écarte le pointeur de la grille. 

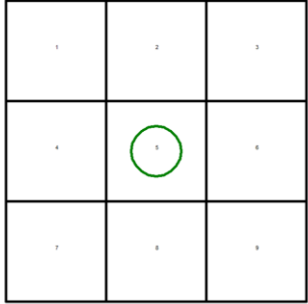
Pour lancer le tracer de la grille, il suffit de taper >>> grille()

Utilitaire pour jouer : un programme pour une croix, un autre pour un rond

Les deux joueurs vont être invités à tracer à tour de rôle soit une croix (pour l'un des joueurs), soit un rond (pour l'autre joueur) sur la grille. Pour identifier la case de la grille, il suffira à chaque joueur de préciser le numéro de ladite case.

Programme	Explications
<pre>def croix(numero): up() goto(Point[numero-1]) down() width(5) color('red') rt(45) fd(70) fd(-140) fd(70) rt(90) fd(70) fd(-140) fd(70) rt(225) up() goto(-350,0) color('black') width(1)</pre>	Comme nous l'avons dit, l'utilisateur devra fournir le numéro de la case choisie. Le curseur va alors se rendre au centre de ladite case puis le stylet va être abaissé puis se déplacer en traçant progressivement une croix rouge. Voici ce que donnera la séquence : <pre>>>> croix(6)</pre> A la fin, on écarte le curseur de la grille. 

Passons à présent au programme pour construire les ronds

Programme	Explications
<pre>>>> def rond(numero): Points = [(-200,150),(0,150),(200,150),(- 200,- 50),(0,-50),(200,-50),(-200,-250),(0,- 250),(200,-250)] width(5) color('green') goto(Points[numero-1]) down() circle(50) up() goto(-350,0) color('black') width(1)</pre>	Comme pour le programme croix, l'utilisateur devra fournir le numéro de la case choisie. Le curseur va alors sur un point précis de ladite case puis le stylet va être abaissé afin de tracer un cercle vert de rayon 50 unités. Voici ce que donnera la séquence : <pre>>>> rond(5)</pre> A la fin, on écarte le curseur de la grille. 

Vous disposez à présent de tous les outils pour jouer à ce jeu intelligent au sens premier du qualificatif intelligent: **percevoir et concevoir**.
Bon jeu!

