

# RAPPORT SAE 1.02

## Table des matières :

<b>I : Gestion des modes de jeu et difficultés :</b>	<b>2</b>
Fonction choix_difficulte :	2
Fonction choix_mode_jeu :	2
Fonction liste_joueur :	3
<b>II : Devinette :</b>	<b>4</b>
Différentes difficultés du bot :	4
Difficulté Hasard :	4
Difficulté Intermédiaire :	4
Difficulté maximum :	4
Jeux d'essais :	5
<b>III : Allumette :</b>	<b>7</b>
Différentes difficultés du bot :	7
Difficulté Hasard :	7
Difficulté Intermédiaire :	7
Difficulté maximum :	7
Jeux d'essais :	7
<b>IV : Morpion :</b>	<b>9</b>
Différentes difficultés du bot :	9
Difficulté Hasard :	9
Difficulté Intermédiaire :	9
Difficulté maximum :	9
Jeux d'essais :	9
<b>V : Puissance 4 :</b>	<b>13</b>
Différentes difficultés du bot :	13
Difficulté Hasard :	13
Difficulté Intermédiaire :	13
Difficulté maximum :	13
Jeux d'essais :	13
<b>VI : Partie Statistique :</b>	<b>16</b>
Devinette :	16
Allumette :	17
Morpion :	17
Puissance 4 :	17

## I : Gestion des modes de jeu et difficultés :

### Fonction choix\_difficulte :

Cette fonction ne prend aucun argument en paramètre, elle n'est constituée que d'un input ainsi que de l'affichage des différentes difficultés. L'input étant la valeur de la difficulté que l'on veut choisir et qui sera renvoyée dans le programme.

L'affichage se trouve sous cette forme :

```
Choisissez le niveau de difficulté de l'IA :
```

```
1 : Hasard / Facile
```

```
2 : Entre-deux / Intermédiaire
```

```
3 : Imbattable
```

```
Entrez le numéro correspondant à la difficulté choisie : █
```

Si l'on choisi 1 pour exemple :

```
Entrez le numéro correspondant à la difficulté choisie : 1
```

```
Difficulté choisie : Hasard / Facile █
```

### Fonction choix\_mode\_jeu :

Cette fonction possède exactement la même structure que celle du choix de la difficulté. Un affichage ainsi qu'un input.

```
Choisissez le mode de jeu :
```

```
1 : Joueur contre Joueur
```

```
2 : Joueur contre IA
```

```
3 : IA contre IA
```

```
Entrez le numéro correspondant au mode de jeu choisi : █
```

Si l'on reprend 1 pour exemple :

```
Entrez le numéro correspondant au mode de jeu choisi : 1
```

```
Mode de jeu choisi : Joueur contre Joueur █
```

### Fonction liste\_joueur :

Cette fonction, elle, a besoin de deux paramètres. Le jeu dans lequel elle se trouve ainsi que du mode de jeu choisi par le joueur ce qui implique d'avoir appelé la fonction choix\_mode\_jeu avant celle-ci.

En fonction du mode de jeu elle va respectivement nous demander ou non de rentrer 1 ou 2 noms de joueurs.

Si 1 comme mode de jeu :

- 2 noms de joueurs à rentrer

Si 2 comme mode de jeu :

- 1 seul nom à rentrer, le deuxième joueur sera automatiquement "IA1"

Si 3 comme mode de jeu :

- Aucun nom à rentrer, les deux joueurs sont automatiquement "IA1" et "IA2"

Comme cela, les noms des IA sont parfaitement définis dès le début.

## II : Devinette :

Devinette ici possède 4 versions :

- Joueur contre Joueur
- Joueur contre IA (Si le joueur choisi et répond à l'IA)
- IA contre Joueur (Si l'IA choisi le nombre et répond au joueur)
- IA contre IA

### Différentes difficultés du bot :

#### Difficulté Hasard :

Si le bot est en difficulté 1 c'est-à-dire le hasard total, il pourra choisir un nombre aléatoire entre 1 et 100 si c'est à l'IA de choisir.

Pour ce qui est du jeu, le bot prendra uniquement en paramètre l'intervalle choisi et jouera entre 1 et l'intervalle jusqu'à tomber sur le bon nombre ce qui peut être très long.

#### Difficulté Intermédiaire :

Si le bot est en difficulté 2 c'est-à-dire l'intermédiaire, il pourra choisir un nombre aléatoire entre 1 et 1000 si c'est à l'IA de choisir.

Pour ce qui est du jeu, le bot prend en paramètre le retour du jeu, le moment où l'autre répond plus grand ou plus petit, la borne maximum qui de base est l'intervalle et la borne minimale qui de base est 1. Ensuite, en fonction de ce qui est répondu, les bornes se resserrent jusqu'à atteindre le nombre mystère. Le bot joue donc aussi aléatoirement entre les bornes mais dans un intervalle qui se réduit.

#### Difficulté maximum :

Si le bot est en difficulté 3 c'est-à-dire le maximum, il pourra choisir un nombre aléatoire entre 1 et 10000 si c'est à l'IA de choisir.

Pour ce qui est du jeu en difficulté maximale, la logique consiste à exploiter les retours du joueur pour réduire l'intervalle de recherche tout en introduisant une stratégie pour augmenter les chances de deviner correctement. Si le joueur indique que le nombre est plus petit, la borne supérieure est ajustée pour être juste en dessous du dernier nombre proposé. Inversement, si le nombre est plus grand, la borne inférieure est relevée à juste au-dessus du dernier nombre. Une fois l'intervalle restreint, si celui-ci est très petit (4 ou moins), le bot choisit un nombre aléatoire dans cet intervalle. Sinon, le bot biaise son choix en favorisant un tiers supérieur ou inférieur de l'intervalle, selon que le nombre est supposé plus grand ou plus petit, ce qui permet une approche plus stratégique et moins aléatoire.

## Jeux d'essais :

Si le joueur choisit le nombre mystère ainsi que l'intervalle :

```
Si vous rentrez autre chose que 'Oui', le compteur sera désactivé.  
Voulez-vous choisir le nombre de tours ? (Oui/Non) : f  
Alex, Choisissez l'intervalle de jeu entre 1 et n : 100  
Alex, Saisissez le nombre mystère :  
Le nombre mystère a bien été enregistré.  
Alex a choisi le nombre à deviner c'est à IA1 de jouer !
```

A chaque itération de l'ia :

```
Le nombre donné par l'IA est 20  
1 : Plus grand  
2 : Plus petit  
3 : C'est gagné !  
Alex, votre choix 1 / 2 / 3
```

Si l'ia a choisit le nombre et l'intervalle :

```
Si vous rentrez autre chose que 'Oui', le compteur sera désactivé.  
Voulez-vous choisir le nombre de tours ? (Oui/Non) : d  
IA1 a choisi le nombre à deviner c'est à Alex de jouer !
```

A chaque itération : (Ne pas prendre en compte le TEMPORAIRE)

```
L'intervalle de jeu est de 1 à 74  
TEMPORAIRE : Le nombre mystère est 45  
Alex, quel est le nombre ?
```

```
Le nombre à deviner n'est pas 5 ! Il est plus grand
```

```
Le nombre à deviner n'est pas 60 ! Il est plus petit
```

Affichage en difficulté 2 et 3 :

Le nombre mystère étant 50 dans notre exemple. Cet affichage est fait pour éclaircir le joueur sur la situation du jeu.

```
Le nombre donné par l'IA est 80 (intervalle: [1, 100])
1 : Plus grand
2 : Plus petit
3 : C'est gagné !
Alex, votre choix 1 / 2 / 3 : 
```

Je répond donc "plus petit" :

```
Le nombre donné par l'IA est 26 (intervalle: [1, 65])
1 : Plus grand
2 : Plus petit
3 : C'est gagné !
Alex, votre choix 1 / 2 / 3 : 
```

Je répond donc "plus grand" :

```
Le nombre donné par l'IA est 49 (intervalle: [27, 65])
1 : Plus grand
2 : Plus petit
3 : C'est gagné !
Alex, votre choix 1 / 2 / 3 : 
```

Et ainsi de suite.

### III : Allumette :

Allumette ici possède 3 versions :

- Joueur contre Joueur
- Joueur contre IA
- IA contre IA

#### Différentes difficultés du bot :

##### Difficulté Hasard :

Dans la première difficulté, le bot joue entièrement aléatoirement et prendra donc 1 à 3 allumettes à chaque tour sans aucune réflexion. Même si sur la fin il ne reste qu'une seule allumette il ne réfléchira pas à prendre le reste, il pourra toujours prendre 2 ou 3, jusqu'à ce qu'il n'en prenne qu'une et finisse la partie.

##### Difficulté Intermédiaire :

Dans la seconde difficulté, c'est divisé entre 2 parties, il a 40% de chance de jouer aléatoirement comme dans la difficulté 1 et il a 60% de chance de jouer de manière optimale qui sera expliquée ci-dessous dans la difficulté suivante.

##### Difficulté maximum :

Dans la difficulté maximale (imbattable), le bot utilise une stratégie optimale pour garantir une victoire. Elle cherche à amener l'adversaire dans une situation perdante, où il ne peut plus gagner. Si le nombre d'allumettes restantes est déjà un multiple de 4 + 1, c'est-à-dire une position perdante pour le bot, elle joue de manière aléatoire, car elle ne peut pas éviter la défaite. Sinon, le bot calcule combien d'allumettes elle doit prendre pour laisser à l'adversaire une quantité d'allumettes équivalente à un multiple de 4 + 1, une position où l'adversaire ne peut que perdre s'il joue correctement. Si cela n'est pas possible (par exemple, si le nombre d'allumettes restantes est trop petit), le bot choisit un nombre d'allumettes au hasard entre 1 et 3.

#### Jeux d'essais :

A chaque itération du bot. L'affichage est évidemment ralenti avec des "sleep" (fonction de la bibliothèque time) pour que le joueur puisse voir ce qu'il se passe sans devoir faire preuve de capacité de lecture très rapide.

```
|-----|  
| Vous êtes dans une partie d'allumettes |  
|-----|
```

```
| Il reste 20 allumettes.
```

```
| C'est au tour de IA1 de jouer.
```

```
| L'IA a pris 3 allumettes
```

Quand c'est au tour du joueur

```
|-----|  
| Vous êtes dans une partie d'allumettes |  
|-----|
```

```
| Il reste 17 allumettes.
```

```
| C'est au tour de Alex de jouer.
```

```
| Alex : Combien d'allumettes voulez-vous prendre ? |
```

Si le bot gagne la partie, il n'a évidemment pas de score inscrit dans les fichiers.

```
| IA1, vous avez perdu car vous avez pris la dernière allumette !
```

```
|-----|
```

```
| Pas de score pour les IA
```



## IV : Morpion :

Morpion ici possède 3 versions :

- Joueur contre Joueur
- Joueur contre IA
- IA contre IA

### Différentes difficultés du bot :

#### Difficulté Hasard :

Dans cette difficulté, le bot sélectionne aléatoirement les coordonnées de la case sur laquelle il va jouer. Il ne possède pas de réflexion lui permettant de gagner via une logique quelconque.

#### Difficulté Intermédiaire :

Dans la seconde difficulté, le bot a deux possibilités de choix de jeu basé sur un pourcentage tiré aléatoirement à chaque tour. Le bot a 1 chance sur 4 de jouer comme le bot en difficulté hasardeuse et le reste de la probabilité lui offre un esprit de réflexion lui permettant d'anticiper les coups qui peuvent, d'abord le faire gagner, et si aucun des coups jouable ne lui permet, il va alors ensuite regarder les coups qui peuvent bloquer la victoire de son adversaire.

#### Difficulté maximum :

En difficulté maximum, le bot n'est plus restreint par des probabilités et possède à tous les coups l'esprit de réflexion lui permettant d'être beaucoup moins aléatoire et de construire une vraie partie ressemblant à celle que l'on pourrait jouer contre un autre humain.

### Jeux d'essais :

#### Cas où le bot est en Hasard :

```
|
  1  2  3
  --- --- ---
1  |  |  |
  ---|---|---
2  |  |  X
  ---|---|---
3  |  |  |
  ---|---|---
Alex, sélectionner la case dans laquelle vous jouez (0) :
```

Le bot joue de manière aléatoire sur la grille et le joueur joue à son tour.

	1	2	3
1	X		
2		O	X
3			

On recommence le processus autant de fois que nécessaire jusqu'à déboucher sur une fin de partie

	1	2	3
1	X	O	X
2	O	O	X
3	X	X	O

Match nul

Cas où le bot est en intermédiaire :

	1	2	3
1	X	X	O
2		O	
3			X

Cet exemplaire sert à montrer que le bot en intermédiaire peut effectivement louper des choses s'il tombe sur la configuration hasardeuse. Dans ce cas de figure précis, le bot a joué en (1,2) alors que le joueur a la colonne pour gagner.

	1	2	3
1	X	X	0
2	X	0	0
3			X

Toujours dans ce cas précis, le joueur décide de ne pas jouer en (3,1) mais en (2,3), le bot va cette fois avoir les bonnes probabilités et voyant qu'il n'a aucune possibilités de gagner, il va bloquer l'une des deux possibilités de victoires étant ici de jouer en (2,1).

	1	2	3
1	X	X	0
2	X	0	0
3	X	0	X

Cas où le bot est en difficulté maximum :

	1	2	3
1			X
2		X	0
3	0		

Dans cette difficulté, le bot réagit à tous mes coups et ne va jamais se tromper ou omettre une possibilité de défense et/ou de victoire

	1	2	3
1	0		X
2		X	0
3	0		X

Deuxième cas où le bot défend une des possibilités de victoire.

	1	2	3
1	0		X
2	0	X	0
3	0	X	X

L'IA a gagné

Dès qu'il en a l'occasion, le bot joue en priorité le coup qui le fait gagner plutôt que de défendre une autre possibilité de victoire pour son adversaire.

## V : Puissance 4 :

Puissance 4 ici possède 3 versions :

- Joueur contre Joueur
- Joueur contre IA
- IA contre IA

### Différentes difficultés du bot :

#### Difficulté Hasard :

Dans cette première difficulté, le bot joue parfaitement aléatoirement parmi les 7 colonnes existantes, malgré tout il ne joue que sur les colonnes non pleines, grâce à la fonction “*colonne\_valide*” et un “*choice*” ensuite.

#### Difficulté Intermédiaire :

Dans la difficulté intermédiaire, le bot joue aléatoirement sauf si il a une occasion de gagner ou d'empêcher la victoire adverse. Il regarde grâce à la fonction “*verif\_victoire\_potentielle*” s'il peut gagner ou contrer la victoire adverse, s'il peut faire les deux il jouera pour sa victoire en priorité.

#### Difficulté maximum :

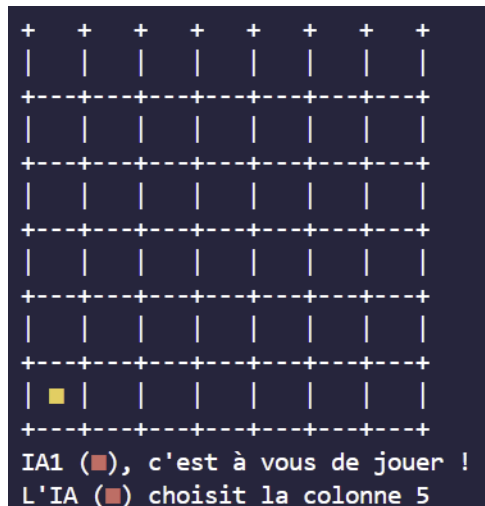
La difficulté maximale n'a pas pu être codée. On voulait qu'elle repose sur la stratégie minimax qui consiste à simuler tous les coups possibles pour les deux joueurs et à choisir le meilleur coup en supposant que l'adversaire jouera de manière optimale.

### Jeux d'essais :

Au tour du joueur :

```
  1  2  3  4  5  6  7
+  +  +  +  +  +  +
|  |  |  |  |  |  |
+---+---+---+---+---+
|  |  |  |  |  |  |
+---+---+---+---+---+
|  |  |  |  |  |  |
+---+---+---+---+---+
|  |  |  |  |  |  |
+---+---+---+---+---+
|  |  |  |  |  |  |
+---+---+---+---+---+
|  |  |  |  |  |  |
+---+---+---+---+---+
Jules (■), c'est à vous de jouer !
Jules (■), choisissez une colonne entre 1 et 7 :
```

Au tour de l'IA :



En difficulté 2 : si l'IA a l'occasion de bloquer la victoire adverse



En difficulté 2 : Si l'IA a l'occasion de gagner



Affichage final :

[illegible]

## VI : Partie Statistique :

Voici le lien du document contenant toutes les statistiques :

[+ STATS SAE](#)

Toutes les statistiques de temps ont été effectuées à l'aide de la fonction "*time*" venant de la bibliothèque éponyme. Et ensuite les moyennes ont été faites à l'aide de la bibliothèque "*numpy*" ainsi que la liste des 10 essais.

### Devinette :

Nous avons essayé chaque difficulté sur 10 coups en bloquant les intervalles de jeu sur 1 à 1000 pour être équivalent sur le nombre de possibilités de nombre mystère.

Nous avons obtenu :

DIFFICULTÉ	MOYENNE BOT QUI DEVINE (secondes)
1	0.0001631498336791992
2	0.00043816566467285155
3	0.00015277862548828124

MOYENNE BOT RÉPONSE (secondes)
0.0000012636184692382812

Efficacité en fonction de la difficulté :

PLUS EFFICACE	ENTRE-DEUX	MOINS EFFICACE
DIFFICULTÉ MAXIMUM	DIFFICULTÉ HASARD	DIFFICULTÉ INTERMÉDIAIRE

Moyenne du nombre de coups par difficulté sur 10 parties :

DIFFICULTÉ	MOYENNE COUP SUR 10 PARTIES
1	385,60
2	12,70
3	12,20



### Allumette :

Comme pour devinette les tests de temps sont établis sur 10 occurrences du bot :

Moyenne de temps en fonction des difficultés :

Difficulté	Moyenne
1	0.00019617080688476561
2	0.0002340078353881836
3	0.0002007007598876953

Ordre d'efficacité par rapport au temps en fonction de la difficulté :

PLUS EFFICACE	ENTRE DEUX	MOINS EFFICACE
DIFFICULTÉ HASARD	DIFFICULTÉ MAXIMUM	DIFFICULTÉ INTERMÉDIAIRE

### Morpion :

Comme pour allumette les tests de temps sont établis sur 10 occurrences du bot :

Moyenne de temps en fonction des difficultés :

DIFFICULTÉ	MOYENNE BOT MORPION
1	0.00006887912750244140
2	0.00009870529174804688
3	0.00009472370147705078

Ordre d'efficacité par rapport au temps en fonction de la difficulté :

PLUS EFFICACE	ENTRE-DEUX	MOINS EFFICACE
DIFFICULTÉ HASARD	DIFFICULTÉ MAXIMUM	DIFFICULTÉ INTERMÉDIAIRE

### Puissance 4 :

Comme pour morpion les tests de temps sont établis sur 10 occurrences du bot :

Moyenne de temps en fonction des difficultés :

DIFFICULTÉ	MOYENNE BOT PUISSANCE 4
1	0.00017378330230712890
2	0.00038805007934570310

Ordre d'efficacité par rapport au temps en fonction de la difficulté :

PLUS EFFICACE	MOINS EFFICACE
DIFFICULTÉ HASARD	DIFFICULTÉ INTERMÉDIAIRE