
SAE 1.02 - Comparaison d'approches algorithmiques

Table des matières

Sujet	3
Expérience utilisateur	3
Devinette	5
Règle du jeu	5
Différentes stratégies	5
Allumette	11
Règle du jeu	11
Différentes stratégies	11
Morpion	17
Règle du jeu	17
Différentes stratégies	17
Outil utilisé	20
Conclusion	20

Sujet

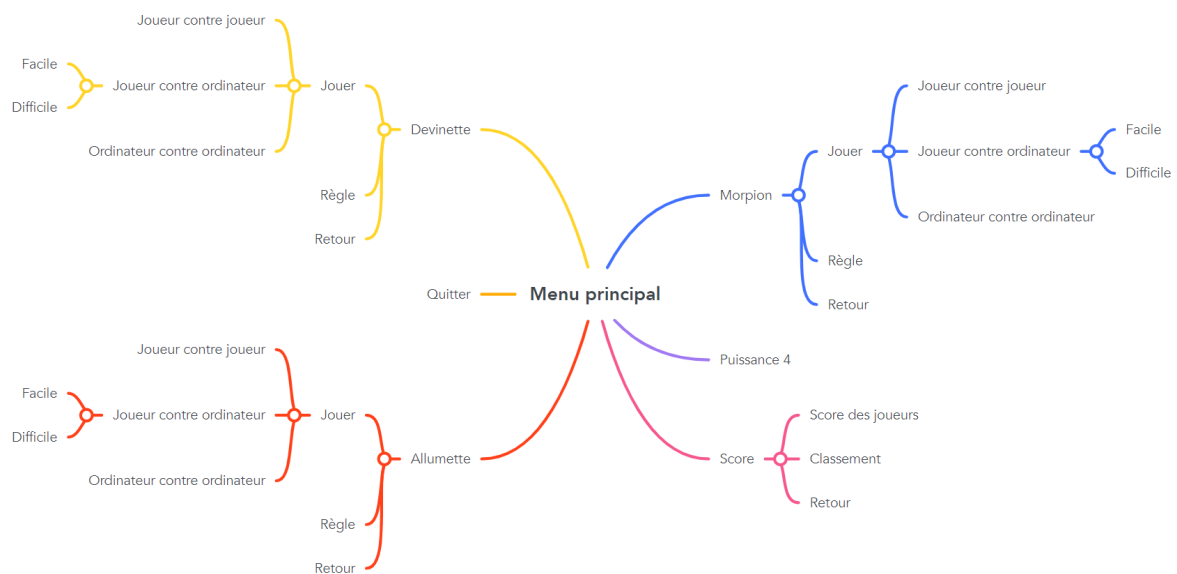
Dans cette SAE il faut reprendre la SAE 1.01 et y ajouter pour chaque jeu les fonctionnalités suivantes :

- Humain contre machine
- Machine contre machine

Pour cela, il existe différentes stratégies du côté machine. Un ordinateur qui opte pour une stratégie totalement hasardeuse ou qui calcule au contraire toutes les possibilités pour choisir la meilleure.

Expérience utilisateur

Pour avoir la meilleure expérience utilisateur possible nous avons repris le même fonctionnement que précédemment mais en rajoutant les options en rapport avec ajout de cette SAE. Ce qui nous donne le schéma suivant.



```
Sous-menu de devinette
1. Jouer
2. Règles
q. Retour au menu principal
Votre choix:
```

Les sous menus de tous les jeux sont gérés par une procédure qui s'adapte en fonction du jeu passé en paramètre.

```
A quel mode jeu souhaitez-vous jouer ?
1. Joueur contre joueur
2. Joueur contre ordinateur
3. Ordinateur contre ordinateur
4. Retour
Votre choix:
```

Le principe est le même pour le choix du mode de jeu.

```
Quel difficulté choisissez-vous ?
1. Facile
2. Difficile
3. Retour
Votre choix:
```

Et une fois de plus pour le choix de la difficulté lorsqu'on choisit le mode de jeu "Joueur contre ordinateur".

Devinette

Règle du jeu

Le joueur 1 choisit un nombre entre 1 et une limite à décider. Le joueur 2 doit deviner ce nombre. A chacune de ses propositions, le joueur 1 répond :

- Trop petit
- Trop grand
- Égale

Différentes stratégies

Pour le jeu Devinette deux niveaux de difficulté sont disponibles.

Facile :

Cette difficulté choisit un nombre aléatoire entre deux bornes. Mini et maxi, et adapte ses bornes en fonction de si le nombre qu'il propose est trop grand ou trop petit.

Algorithme :

```
fonction ordiFacile(mini: entier, maxi: entier) retourne entier:
debut
    """
    fonction qui permet à l'ordinateur de choisir un nombre entre
mini et maxi
    Args:
        mini: minimum
        maxi: maximum

    Returns: le nombre choisi par l'ordinateur
    """
    retourne random(mini, maxi)
fin ordiFacile
```

Delhomel Mattéo
Pinaud Louis

Jeux d'essais:

Le nombre à trouver est "7".

```
DELMAT entrez la borne superieure à 1 : 10  
Entrez le nombre que DELMAT doit trouver entre 1 et 10 :
```

```
L'ordinateur à choisit 3 à vous de jouer  
3 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
DELMAT faites votre choix : 2
```

```
L'ordinateur à choisit 4 à vous de jouer  
4 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
DELMAT faites votre choix : 2
```

```
L'ordinateur à choisit 7 à vous de jouer  
7 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
DELMAT faites votre choix : 3
```

```
L'ordinateur à trouvé le nombre en 3 coups  
DELMAT à vous de trouver le nombre que l'ordinateur a choisis entre 1 et 10  
DELMAT entrez un nombre :
```

```
5 est trop petit  
DELMAT entrez un nombre :
```

```
7 est trop petit  
DELMAT entrez un nombre :
```

```
DELMAT à trouvé le nombre en 3 coups  
Match nul
```

Delhomel Mattéo
Pinaud Louis

Difficile :

Cette difficulté repose sur un algorithme qui fonctionne sur le même principe qu'une recherche dichotomique. C'est-à-dire que l'on choisit le nombre qui se situe entre deux bornes et l'on change ses bornes en fonction de si le nombre est trop grand ou trop petit.

Algorithme :

```
fonction ordiDifficile(mini: entier, maxi: entier) retourne entier:
debut
    """
    fonction qui permet à l'ordinateur de choisir un nombre entre
mini et maxi
    Args:
        mini: borne inférieur
        maxi: borne supérieur
    Retourne:
        un nombre entre mini et maxi
    """

    Avec n: entier
    n <- (maxi - mini) / 2 + mini
    retourne n
fin ordiDifficile
```

Delhomel Mattéo
Pinaud Louis

Jeux d'essais:

Le nombre a trouver est 7.

```
delmat entrez la borne superieure à 1 : 10  
Entrez le nombre que delmat doit trouver entre 1 et 10 :
```

```
L'ordinateur à choisit 5 à vous de jouer  
5 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
delmat faites votre choix : 2
```

```
L'ordinateur à choisit 8 à vous de jouer  
8 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
delmat faites votre choix : 1
```

```
L'ordinateur à choisit 6 à vous de jouer  
6 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
delmat faites votre choix : 2
```

```
L'ordinateur à choisit 7 à vous de jouer  
7 est :  
1. Trop grand  
2. Trop petit  
3. Egal  
delmat faites votre choix : 3
```

```
L'ordinateur à trouvé le nombre en 4 coups  
delmat à vous de trouver le nombre que l'ordinateur a choisis entre 1 et 10  
delmat entrez un nombre :
```

L'avantage de cet algorithme est qu'il ne repose pas sur la chance comme le précédent, mais sur des calculs. Mais cet avantage peut rendre plus lent dans certains cas. Par exemple avec des plages de nombre plus petite.

Les jeux d'essai ont été réalisés avec le mode de jeu "Joueur contre ordinateur".

Delhomel Mattéo
Pinaud Louis

Ordinateur contre ordinateur:

Pour le mode de jeu "Ordinateur contre ordinateur" on fait s'affronter les deux ordinateurs présentés ci-dessus.

Le joueur "ordinateur 1" utilise l'algorithme de l'ordinateur en mode facile. Et le joueur "ordinateur 2" utilise celui de l'ordinateur en mode difficile.

Pour comparer les deux algorithmes nous allons les tester de deux manières différentes.

- Une première partie avec une petite plage de nombre entre 1 et 10.
- Une deuxième partie dans une plage de nombre beaucoup plus large entre 1 et 100.
- Dans les deux parties les nombres à trouver seront 7 et 62.

Jeux d'essais:

Première partie

Ordinateur 1 :

```
Au tour de l'ordinateur 1
L'ordinateur à choisit 8
8 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 7
Appuyez sur "Entrée" pour continuer :
```

```
4 est trop petit
Appuyez sur "Entrée" pour continuer :
```

Ordinateur 2 :

```
L'ordinateur 1 à trouvé en 3 coups
Au tour de l'ordinateur 2
L'ordinateur à choisit 5
5 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 6
6 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 8
8 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 7
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur 2 à trouvé en 4 coups
L'ordinateur 1 à gagné en 3 coups
Appuyez sur "Entrée" pour continuer :
```

Delhomel Mattéo
Pinaud Louis

Deuxième partie :
Ordinateur 1 :

```
Au tour de l'ordinateur 1
L'ordinateur à choisit 96
96 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 72
72 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 8
8 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 28
28 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 61
61 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 63
63 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 62
Appuyez sur "Entrée" pour continuer :
```

Ordinateur 2 :

```
L'ordinateur 1 à trouvé en 7 coups
Au tour de l'ordinateur 2
L'ordinateur à choisit 50
50 est trop petit
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 75
75 est trop grand
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur à choisit 62
Appuyez sur "Entrée" pour continuer :
```

```
L'ordinateur 2 à trouvé en 3 coups
L'ordinateur 2 à gagné en 3 coups
Appuyez sur "Entrée" pour continuer :
```

Grâce à ces deux parties, on remarque que l'ordinateur qui repose ses réponses sur des calculs a plus de chance de l'emporter sur des plages de nombre plus élevé.

Allumette

Règle du jeu

On dispose d'un tas de 20 allumettes. Chaque joueur à tour de rôle peut prélever 1, 2 ou 3 allumettes. Le perdant est celui qui pioche la dernière allumette.

Différentes stratégies

Tout comme le jeu Devinette, deux niveaux de difficulté sont disponibles.

Facile :

Cette difficulté repose sur de l'aléatoire. C'est à dire que tant que le nombre d'allumette est supérieure 3, l'ordinateur choisit un nombre aléatoire de d'allumette à retirer entre 1 et 3. Mais si le nombre d'allumette restante est inférieur ou égal à 2, l'ordinateur va choisir un nombre entre 1 et le nombre d'allumette restante. Ceci permet d'éviter de retirer plus d'allumette qu'il n'en reste en jeu.

Algorithme :

```
fonction ordiFacile(nb_allumette) retourne Entier:
debut
    """Choisit combien d'allumettes seront retirées par le bot niveau
facile
    Args:
        nb_allumette: Nombnre d'allumettes restantes

    Retourne:
        Entier choisi par le bot
    """
    Si nb_allumette >= 3 Alors
        retourne randint(1, 3)
    sinon
        si nb_allumette <= 2 Alors
            retourne randint(1, nb_allumette)
        finsi
    finsi
fin ordiFacile
```

Delhomel Mattéo

Pinaud Louis

Jeux d'essais :

```
| | | | | | | | | | | | | | | | | |  
  
Il reste 20 allumettes  
C'est au tour de l'ordi de jouer  
  
| | | | | | | | | | | | | | | | | |  
  
Il reste 19 allumettes  
C'est à votre tour de jouer  
delmat, retirez entre 1 et 3 allumettes : 3
```

```
| | | | | | | | | | | | | | | | | | . . . .  
  
Il reste 16 allumettes  
C'est au tour de l'ordi de jouer  
  
| | | | | | | | | | | | | | | | | | . . . .  
  
Il reste 15 allumettes  
C'est à votre tour de jouer  
delmat, retirez entre 1 et 3 allumettes : 2
```

```
| | | | | | | | | | | | | | | | | | . . . . .  
  
Il reste 13 allumettes  
C'est au tour de l'ordi de jouer  
  
| | | | | | | | | | | | | | | | | | . . . . .  
  
Il reste 11 allumettes  
C'est à votre tour de jouer  
delmat, retirez entre 1 et 3 allumettes : 3
```

```
| | | | | | | | | | | | | | | | | | . . . . .  
  
Il reste 8 allumettes  
C'est au tour de l'ordi de jouer  
  
| | | | | | | | | | | | | | | | | | . . . . .  
  
Il reste 5 allumettes  
C'est à votre tour de jouer  
delmat, retirez entre 1 et 3 allumettes : 3
```

```
| | . . . . . . . . . . . . . . . . . . . .  
  
Il reste 2 allumettes  
C'est au tour de l'ordi de jouer  
  
| . . . . . . . . . . . . . . . . . . . .  
  
Il reste 1 allumettes  
C'est à votre tour de jouer  
delmat, retirez entre 1 et 3 allumettes : 1
```

```
L'ordinateur à gagné  
Appuyez sur "Entrée" pour continuer : 
```

Difficile :

Pour l'ordinateur difficile, nous avons utilisé une solution mathématique. Pour gagner au jeu des Allumettes l'objectif est que notre adversaire se retrouve à piocher des allumettes lorsqu'il en reste 5 pour être sûr de l'emporter. Mais pour être sûr de réunir cette condition dès le début de la partie, il suffit de faire en sorte que notre adversaire se retrouve à piocher des allumettes lorsqu'il reste un nombre d'allumette qui soit un multiple de 4 plus 1. Par exemple 17, 13, 9 et 5.

Pour ce faire, si l'on commence à jouer, il suffit de piocher 3 allumettes dès le début. Si ce n'est pas le cas, nous avons choisi de faire piocher une allumette à l'ordinateur jusqu'à qu'il est l'occasion d'obtenir un multiple de 4 plus 1.

Algorithme :

```
fonction ordiDifficile(nb_allumette) retourne Entier:
début
    ""Choisit combien d'allumettes seront retirées par le bot niveau
    difficile

    Args:
        nb_allumette: Nombnre d'allumettes restantes

    Retourne:
        Entier choisi par le bot
    ""
    Si nb_allumette = 20 Alors
        retourne 3
    sinon
        Si 1 < nb_allumette et nb_allumette <= 4 Alors
            retourne nb_allumette - 1
        sinon
            pour i allant de 1 à 3 faire
                si ((nb_allumette - i) % 4) = 1 alors
                    retourne i
                finSi
            finFaire
        retourne 1
    finSi
finSi
```

Delhomel Mattéo

Pinaud Louis

Jeux d'essais :

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

C'est au tour de l'ordi de jouer

| | | | | | | | | | | | | | | | | |
Il reste 17 allumettes

C'est à votre tour de jouer
delmat, retirez entre 1 et 3 allumettes : 1
```

```
| | | | | | | | | | | | | | | | | |
Il reste 16 allumettes

C'est au tour de l'ordi de jouer

| | | | | | | | | | | | | | | | | |
Il reste 13 allumettes

C'est à votre tour de jouer
delmat, retirez entre 1 et 3 allumettes : 3
```

```
| | | | | | | | | | | | | | | | | |
Il reste 10 allumettes

C'est au tour de l'ordi de jouer

| | | | | | | | | | | | | | | | | |
Il reste 9 allumettes

C'est à votre tour de jouer
delmat, retirez entre 1 et 3 allumettes : 2
```

```
| | | | | | | | | | | | | | | | | |
Il reste 7 allumettes

C'est au tour de l'ordi de jouer

| | | | | | | | | | | | | | | | | |
Il reste 5 allumettes

C'est à votre tour de jouer
delmat, retirez entre 1 et 3 allumettes : 1
```

```
| | | | | | | | | | | | | | | | | |
Il reste 4 allumettes

C'est au tour de l'ordi de jouer

| | | | | | | | | | | | | | | | | |
Il reste 1 allumettes

C'est à votre tour de jouer
delmat, retirez entre 1 et 3 allumettes : 1
```

L'ordinateur à gagné

Appuyez sur "Entrée" pour continuer :

Delhomel Mattéo
Pinaud Louis

Ordinateur contre ordinateur :

Le mode de jeu "Ordinateur contre ordinateur" fait s'affronter les ordinateurs en mode facile et difficile. Tout en sachant que l'ordinateur "facile" joue toujours en premier.

Jeux d'essais:

```
| | | | | | | | | | | | | | | |
Il reste 20 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | | | | | | | | | | | .
Il reste 18 allumettes
C'est au tour du bot 2 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | | | | | | | | | | . .
Il reste 17 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | | | | | | | | | | . . .
Il reste 16 allumettes
C'est au tour du bot 2 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | | | | | | | . . . . .
Il reste 13 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer :
```

```
| | | | | | | | | | . . . . . .
Il reste 11 allumettes
C'est au tour du bot 2 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | | | | . . . . . . . .
Il reste 9 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | . . . . . . . . . .
Il reste 6 allumettes
C'est au tour du bot 2 de jouer
Appuyez sur "Entrée" pour continuer : █
```

```
| | | | | . . . . . . . . . .
Il reste 5 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer :
```

```
| | . . . . . . . . . . . . .
Il reste 2 allumettes
C'est au tour du bot 2 de jouer
Appuyez sur "Entrée" pour continuer :
```

```
| . . . . .
```

Il reste 1 allumettes
C'est au tour du bot 1 de jouer
Appuyez sur "Entrée" pour continuer :

Bot 2 à gagné
Appuyez sur "Entrée" pour continuer :

Grâce au mode de jeu “ordinateur contre ordinateur”, nous pouvons comparer les deux algorithmes. Grâce à ça, nous pouvons voir que l'algorithme qui utilise des calculs est beaucoup plus efficace. dès le premier tour après que l'ordinateur facile ait joué il se met déjà en condition pour gagner puisqu'après son tour il reste 17 allumettes soit un multiple de 4 plus 1 et ceux jusqu'à la fin de la partie.

Morpion

Règle du jeu

On dispose d'un plateau de 3 cases par 3 cases. Chaque joueur pose tour à tour un pion X ou O sur la grille. Le premier à aligner 3 pions de la même sorte gagne la partie. En cas de plateau rempli, il y a égalité.

Différentes stratégies

Ici encore, deux niveaux de difficulté existent. Cependant, l'algorithme sera le même pour les deux difficultés.

Le bot utilisera la série de conditions suivante pour choisir la bonne case dans laquelle jouer:

SI	ALORS
Case centrale libre	Jouer dans la case centrale
Une ligne / colonne a une case libre et deux cases avec le pion allié	Jouer dans la case libre
Une ligne / colonne a une case libre et deux cases avec le pion ennemi	
Coin libre	Jouer sur le dernier coin libre

On fait en sorte de séparer les vérifications alliées et ennemies, pour permettre de prioriser sa propre victoire plutôt que le blocage d'un coup ennemi.

Selon ces conditions, il est normalement impossible pour le bot de perdre si l'algorithme s'exécute totalement, il forcera au moins l'égalité. Cependant, le but de notre bot n'est pas d'être invincible, mais de remplacer un joueur, qui peut commettre des erreurs. Nous ajoutons donc un taux de précision, qui diffère entre les bots faciles et difficiles.

On tirera un entier entre 1 et 100 qui, si il est supérieur à sa précision, l'empêchera de vérifier que son coup est le meilleur :

```
Avec precision : entier

# Le bot sélectionne une case aléatoire sur laquelle jouer ...

Si randint(1,100) < precision alors:

    # On effectue le choix grâce aux conditions de l'algo précédent

Fin Si
retourne case choisie
```

On fait choisir au programme une case aléatoire, qui sera remplacée par le choix le plus pertinent si l'on tombe dans sa précision avec le randint(). En clair, un bot avec une précision de "80" a 20% de chance de jouer sur une case aléatoire plutôt que de jouer le meilleur coup.

Il est ainsi possible de battre chacun des bots, grâce à la possibilité qu'ils commettent une erreur. Nous avons fixé la précision facile à 40% des coups, et la précision difficile à 80%.
Jeux d'essais :

```
  0 1 2
0 X - -
1 - - -
2 - - -
  0 1 2
0 X - -
1 - - -
2 0 - -
delmat entrez la ligne :
```

Ici le bot manque son coup, il est censé jouer au centre en priorité

```
  0 1 2
0 X - -
1 - - -
2 0 - X
  0 1 2
0 X - -
1 - 0 -
2 0 - X
delmat entrez la ligne :
```

Ici le bot joue correctement, il place son pion au centre

```
  0 1 2
0 X - X
1 - 0 -
2 0 - X
  0 1 2
0 X - X
1 - 0 0
2 0 - X
delmat entrez la ligne :
```

Ici le bot joue également correctement, en bloquant la victoire du joueur

```

  0 1 2
0 X X X
1 - 0 0
2 0 - X

delmat a gagné !

Appuyez sur "Entrée" pour continuer :

```

Le joueur s'impose finalement, car il a réussi à rendre impossible sa défaite en préparant deux coups. Etant donné que le bot cherche les coins en priorité, seul son erreur est à l'origine de sa défaite.

Le bot testé ci-dessus avait 40% de chances de jouer le bon coup. Le principe est le même pour toutes les précisions, si ce n'est que la fréquence des erreurs change.

On peut désormais comparer les deux bots en les faisant jouer l'un contre l'autre. L'un est un bot facile à 40% de précision, l'autre un bot difficile avec 80%.

```

  0 1 2
0 - - -
1 - - -
2 - - -

  0 1 2
0 - - -
1 - X -
2 - - -

  0 1 2
0 - - -
1 - X X
2 - - -

  0 1 2
0 - 0 -
1 0 X X
2 - - -

  0 1 2
0 - 0 0
1 0 X X
2 - - X

  0 1 2
0 - 0 0
1 0 X X
2 - X X

  0 1 2
0 - 0 0
1 0 X X
2 0 X X

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

```

Sur les exemples ci-dessus, le bot difficile joue avec les croix, et le bot facile avec les ronds. On constate à la fin de la partie la victoire du bot difficile. Seulement, chacun des deux bots peut être amené à perdre une partie sur une erreur, en fonction de la chance. Pour pouvoir analyser ces données, on utilise un tableau de stats basé sur 50 parties.

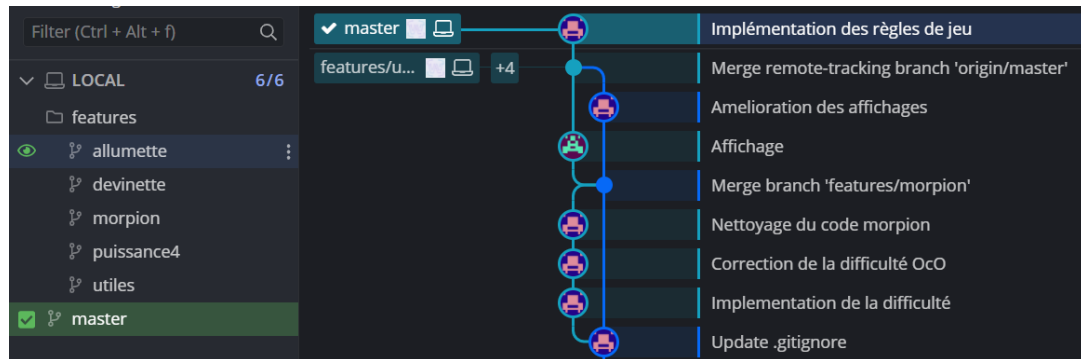
DIFFICULTÉ BOT	% DE VICTOIRES	% D'ÉGALITÉ
FACILE	7,5 %	22,5 %
DIFFICILE	70 %	

On remarque que chacun des deux bots peut être gagnant d'une partie, ou qu'une égalité peut avoir lieu. Cependant, le bot difficile reste bien plus performant que son homologue.

Outil utilisé

Pour réaliser cette SAE nous avons utilisé les outils qui nous ont paru les plus simples et les plus pratiques possible. Pour cela nous avons utilisé les outils suivant :

- [github](#) pour stocker et versionner notre code et modifier notre code sans gêner l'autre à l'aide de branche.
- Git kraken pour gérer notre dépôt à l'aide d'une interface graphique.



Conclusion

En définitive, cette SAE est très intéressante car elle permet de faire un vrai projet plus long qui requiert plus d'investissement et avec plus de contenu que pendant les tp, ce qui est plus représentatif de ce que l'on peut trouver dans le monde du travail.