

TP - Séance n°2

Implémentation du jeu de l'Abapa

Si vous n'avez pas fait la partie facultative du dernier TD, commencez par lire *entièrement* ce sujet de TP. Avant de vous mettre à programmer, réfléchissez aux classes qu'il faut définir, et choisissez pour chaque méthode la classe dans laquelle vous allez l'écrire. Vous trouverez quelques suggestions à la fin du sujet. Évidemment, il y a plusieurs manières de faire, mais certaines solutions sont plus judicieuses que d'autres. Si vous avez un doute, parlez-en à votre enseignant.

1 Jeu de l'Abapa

Le jeu de l'Abapa se joue à deux sur un plateau composé de deux rangées de 6 trous. Chaque joueur possède une rangée de 6 trous, plus un trou (appelé kalah) situé à droite de sa rangée.

Au départ, on répartit quarante-huit graines dans les douze trous (kalahs exclues) à raison de quatre graines par trou. Les joueurs sont l'un en face de l'autre, avec une rangée devant chaque joueur. Cette rangée sera son camp. On choisit un sens de rotation qui vaudra pour toute la partie (sens inverse des aiguilles d'une montre pour les dessins). On choisit également un joueur qui commencera la partie. Un tour se joue de la façon suivante : le premier joueur prend toutes les graines d'un des trous de son camp puis il les égraine dans toutes les cases qui suivent ce trou sur sa rangée puis sur celles de son adversaire suivant le sens de rotation, y compris dans les kalahs. Par exemple, dans la figure 1, le joueur 1 a vidé le 2ème trou en partant de la droite et a distribué son contenu dans les 4 trous suivants (kalah comprise).

Si sa dernière graine tombe dans un trou du camp adverse et qu'il y a alors deux ou trois graines dans ce trou, le joueur récupère ces deux ou trois graines et les met dans sa kalah. Ensuite, il regarde la case précédente : si elle est dans le camp adverse et contient deux ou trois graines, il récupère ces graines, et ainsi de suite jusqu'à ce qu'il arrive à son camp ou jusqu'à ce qu'il y ait un nombre de graines différent de deux ou trois.

On ne saute pas de case lorsqu'on égraine sauf lorsqu'on a plus de quatorze graines, c'est-à-dire qu'on fait un tour complet : on ne remplit pas la case où l'on vient de prendre les graines. Il faut nourrir l'adversaire, c'est-à-dire que, quand celui-ci n'a plus de graines, il faut absolument jouer un coup qui lui permette de rejouer ensuite. Si ce n'est pas possible, la partie s'arrête et le joueur qui allait jouer capture les graines restantes. Si un coup devait prendre toutes les graines

adverses, alors le coup peut être joué, mais aucune capture n'est faite : il ne faut pas affamer l'adversaire.

La partie s'arrête quand un des joueurs a capturé au moins 25 graines, soit plus de la moitié.

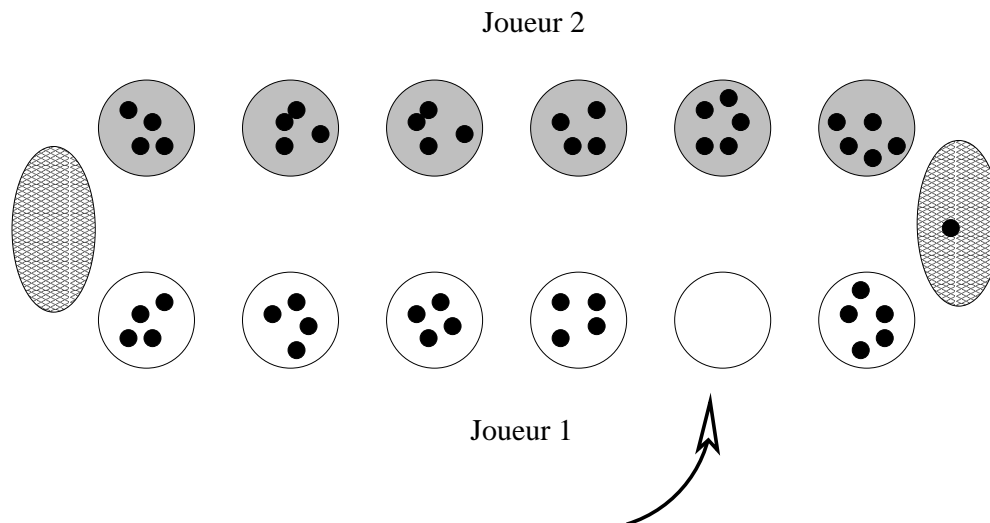


FIGURE 1 – Le joueur 1 vient de jouer son premier coup.

2 Quelques pistes

Si vous ne savez pas comment attaquer le problème, voici quelques suggestions :

- Pour la représentation visuelle du plateau, contentez-vous d'un affichage texte du genre :

```

      4   4   4   4   5   5
0      4   4   4   4   0   5      1

```

- Pour la représentation interne du plateau, vous pouvez définir une classe **Plateau** dont chaque instance contient une référence vers un tableau d'entiers de taille 14. La situation de la figure 1 pourrait, par exemple, être représentée par le tableau

```
[4, 4, 4, 4, 0, 5, 1, 5, 5, 4, 4, 4, 4, 0],
```

qui commence par la première case du joueur 1 et suit le sens de rotation.

- Afin de limiter l'accès au plateau, une idée serait d'ajouter à **Plateau** une méthode **jouer** qui reçoit l'identifiant d'un trou en argument et s'occupe de distribuer les graines de ce trou dans les cases suivantes, et, le cas échéant, de capturer les graines de l'adversaire. (Il est peut-être plus naturel d'identifier

un trou à l'aide d'une paire $(i, j) \in \{0, 1\} \times \{0, \dots, 6\}$ au lieu d'un seul nombre $i \in \{0, \dots, 13\}$.) La méthode `jouer` vérifie aussi que les règles du jeu de l'Abapa sont respectées, et refuse de jouer une action interdite. (N'hésitez pas à découper le travail en plusieurs fonctions auxiliaires.)

- Chaque joueur peut être représenté par un objet du type `Joueur`, avec une méthode `actionChoisie` qui retourne le numéro du trou que le joueur veut vider. (Comme chacun peut seulement jouer dans son propre camp, on peut spécifier ce trou par un nombre entre 0 et 6.) Dans un premier temps, vos joueurs seront des humains, donc il suffit de lire le numéro à partir du clavier.
- Construisez un objet du type `Jeu` qui contient des références vers le plateau et les deux joueurs, ainsi qu'un entier `actif` pour savoir lequel des joueurs est le prochain à jouer. L'idée est de décomposer votre programme en plusieurs méthodes de la classe `Jeu`, ce qui permet d'avoir une fonction `main` propre et facile à comprendre. Par exemple :

```
class Abapa {
    public static void main(String[] args) {
        Jeu jeu = new Jeu();
        jeu.afficherPlateau();
        while (!jeu.fini()) {
            jeu.jouerTour();
            jeu.afficherPlateau();
        }
        jeu.afficherGagnant();
    }
}
```

- Si vous avez du temps, vous pouvez vous amuser à développer une « intelligence artificielle ». (Ça peut être tout bêtement un joueur qui joue au hasard.) Une manière assez naturelle de l'intégrer dans le programme serait de dériver une sous-classe `JoueurArtificiel` de la classe `Joueur`. Dans ce cas, il serait bien de déplacer le code propre au joueur humain dans une autre sous-classe `JoueurHumain`.