

# Implémentation d'un besoin client

M.Adam – N.Delomez – JF.Kamp – L.Naert

3 octobre 2022

## Objectifs de la SAÉ

- Développer des applications informatiques simples
- **AC 1** Implémenter des conceptions simples
- **AC 3** Faire des essais et évaluer leurs résultats en regard des spécifications

## Jeu de Marienbad

Ce jeu de société combinatoire abstrait, dont il existe plusieurs variantes, se joue avec des graines, des dominos, des jetons, des cartes, des allumettes, ... Il appartient à la famille plus large des jeux de Nim utilisé dans Ford Boyard.

Cette variante a été rendue célèbre par le film d'Alain Resnais "L'Année dernière à Marienbad".

Il y a quatre rangées, avec respectivement 1, 3, 5, 7 allumettes. À chaque tour, le joueur prend le nombre d'allumettes qu'il veut, au moins une et dans une même rangée. Le joueur qui prend la dernière allumette gagne.

Une version en ligne du jeu est disponible sur <http://therese.eveilleau.pagesperso-orange.fr/>.

L'objectif de cette SAÉ est d'écrire un programme java qui permet de jouer des parties du jeu de Marienbad.

## Partie 1 : joueur contre joueur

Dans une premier temps, le programme doit permettre à deux joueurs de jouer l'un contre l'autre.

### Déroulement d'une partie

Une partie se déroule souvent le scénario suivant :

1. saisie du nom du joueur qui joue en premier,
2. saisie du nom du joueur qui joue en deuxième,

3. affichage du jeu :

```

0 : |
1 : | | |
2 : | | | | |
3 : | | | | | | |

```

4. affichage du nom du joueur qui joue,
5. saisie de la ligne et du nombre d'alumettes à enlever,
6. changement de joueur,
7. reprise de la séquence d'affichage du jeu et du nom du joueur, de saisie de la ligne et du nombre d'alumettes,
8. arrêt du jeu quand il ne reste plus aucune alumette,
9. affichage du nom du vainqueur.

Le programme doit comporter plusieurs méthodes. Chaque méthode doit posséder sa méthode de test. Il doit être facile de passer à une version avec un autre nombre de lignes.

## A rendre

- un fichier pdf expliquant la stratégie employée par le programme pour jouer contre le joueur,
- le fichier source : Marienbad.java,
- la trace de l'exécution des méthodes de test,
- la trace de l'exécution d'une partie.

Les trois fichiers doivent être compressés dans une archive Nom\_Prénom\_Partie1.zip, par exemple Le-Guen\_Olivier\_Partie1.zip.

## Partie 2 : joueur contre ordinateur

### Stratégie gagnante

La stratégie gagnante est expliquée sur <https://fr.wikipedia.org/>.

La méthode repose sur le système binaire. La position de départ, précisée par le dessin ci-dessous, s'analyse à l'aide des calculs suivants :

```

1 = 0 0 1    en binaire
3 = 0 1 1      "
5 = 1 0 1      "
7 = 1 1 1      "

```

Si on effectue les sommes des chiffres du binaire colonne par colonne en base dix, on trouve :

```

S = 2 2 4

```

Considérons la variante où le gagnant est celui qui prend la dernière allumette. Selon le théorème de Sprague-Grundy, une position est gagnante pour le joueur qui l'atteint si et seulement si tous les chiffres de  $S$  sont pairs. Une telle position sera perdante pour le joueur qui part d'une telle position. Ainsi, dans l'exemple donné, la position initiale est perdante pour le premier joueur, son adversaire ayant la possibilité de conserver cette propriété de  $S$  tout le long de la partie jusqu'à ce qu'il ne reste plus d'allumette.

## Déroulement d'une partie

Une partie se déroule souvent le scénario suivant :

1. saisie du nom du joueur,
2. choix de celui qui joue en premier joueur ou ordinateur,
3. affichage du jeu :

```
0 : |  
1 : | | |  
2 : | | | | |  
3 : | | | | | | |
```

4. affichage du nom du joueur qui joue,
5. saisie de la ligne et du nombre d'allumettes à enlever s'il s'agit du joueur.
6. changement de joueur,
7. reprise de la séquence d'affichage du jeu et du nom du joueur, de saisie de la ligne et du nombre d'allumettes,
8. arrêt du jeu quand il ne reste plus aucune allumette,
9. affichage du nom du vainqueur.

## A rendre

- Un fichier pdf décrivant la structure utilisée pour stocker le jeu,
- le fichier source : Marienbad.java,
- la trace de l'exécution des méthodes de test,
- la trace de l'exécution d'une partie.

Les trois fichiers doivent être compressés dans une archive Nom\_Prenom\_Partie2.zip, par exemple Le-Guen\_Olivier\_Partie2.zip.

## Critères d'évaluation

La SAÉ sera évaluée suivant les critères suivants :

- respect du format du rendu, nom du fichier, et compression en zip,

- respect de la date de rendu,
- respect des conventions java,
- décomposition en méthodes,
- méthodes de test,
- exécution correcte et sans erreur du programme,
- qualité du code rendu.

Cette liste n'est pas exhaustive et d'autres critères pourront être ajoutés au choix par les enseignants de la SAÉ.