



**UFR d'Informatique**

Université Paris Diderot - UFR Informatique  
Bâtiment Sophie Germain, 8 place Aurélie Nemours  
UFR Informatique 3ème étage, 75013 Paris  
Tél. +33 (0)1 57 27 68 90  
[www.informatique.univ-paris-diderot.fr](http://www.informatique.univ-paris-diderot.fr)

## **Master 1 Informatique 2016 - 2017**

### **Projet M1LOAC**

## **Framework**

#### **Enseignants**

Olivier Carton  
[Olivier.Carton@liafa.univ-paris-diderot.fr](mailto:Olivier.Carton@liafa.univ-paris-diderot.fr)  
Yan Jurski  
[Yan.Jurski@liafa.univ-paris-diderot.fr](mailto:Yan.Jurski@liafa.univ-paris-diderot.fr)  
Isabelle Fagnot  
[fagnot@univ-paris-diderot.fr](mailto:fagnot@univ-paris-diderot.fr)  
khaled maamra

#### **Étudiants**

Binta Diabira  
[binta.diabira@etu.univ-paris-diderot.fr](mailto:binta.diabira@etu.univ-paris-diderot.fr)  
Meriem FEKIH AHMED  
[fekihahm@informatique.univ-paris-diderot.fr](mailto:fekihahm@informatique.univ-paris-diderot.fr)

Master1 Informatique 2016 - 2017

Université Paris Diderot - UFR Informatique

Version du 2 janvier 2017



# Table des matières

---

<b>1</b>	<b>Framework : 2048 et Taquin</b>	<b>4</b>
1.1	Contenu du rendu . . . . .	4
1.2	La modélisation . . . . .	4
1.3	Les classes . . . . .	5
1.3.1	Classe Jeux<T> . . . . .	5
1.3.2	Classe Case<T> . . . . .	5
1.3.3	Les classes Taquin, jeu2048, et sokoban . . . . .	5
1.3.4	Jeu2048Negative . . . . .	5
1.3.5	Jeu2048TwoThreeFive . . . . .	5
1.3.6	Pour le sokoban(non implémenté) . . . . .	5
1.4	Exécution du jeu . . . . .	5
1.4.1	make . . . . .	5
1.4.2	./Jeux . . . . .	5
1.4.3	make clean . . . . .	5

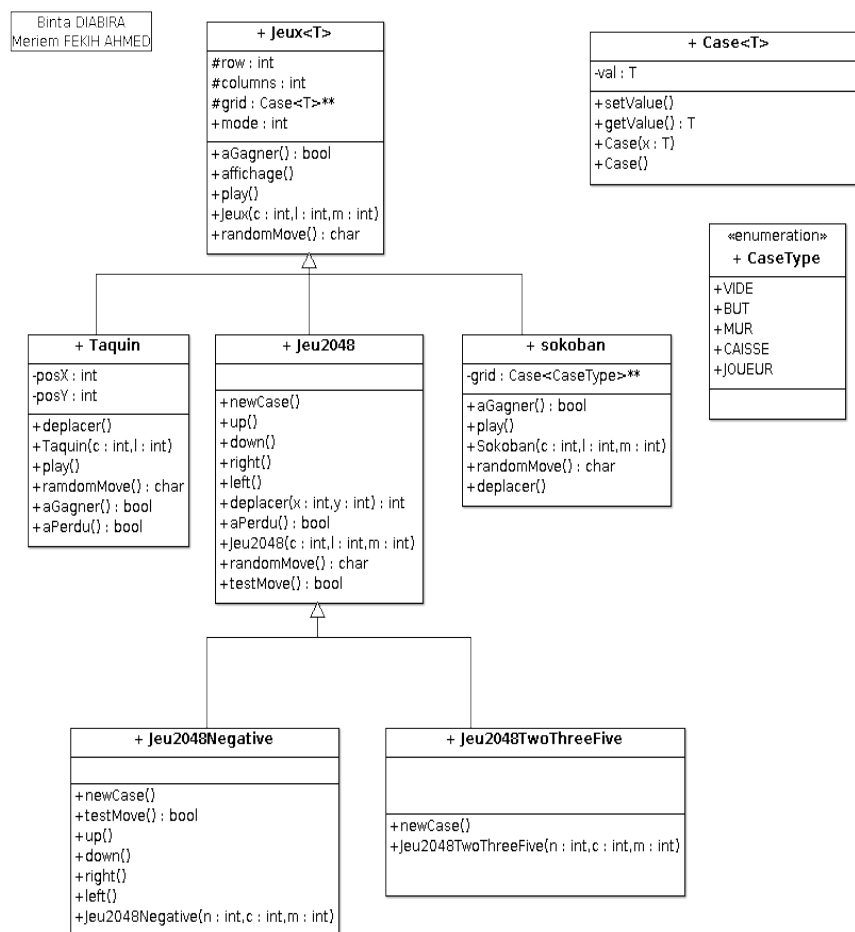
# Framework : 2048 et Taquin

---

## 1.1 Contenu du rendu

- Diagramme de classes
- Fichiers sources : Classes, bibliothèques ...
- Makefile : pour compiler le code et exécuter les jeux en mode terminal.
- Un rapport expliquant les parties traitées et les pistes d'extension implémentées.
- Un Readme expliquant comment exécuter le jeu.

## 1.2 La modélisation



## 1.3 Les classes

### 1.3.1 Classe Jeux<T>

- Classe servant de base pour les différents jeux à implémenter.
- L'attribut grid est un tableau de pointeur de Case représentant la grille de jeu.
- Les attributs row et columns sont les dimension du plateau de jeu.

### 1.3.2 Classe Case<T>

Classe représentant les cases de la grille des différents jeux, chaque case à une valeur dont le type diffère selon le jeu(entier ou enumeration).

### 1.3.3 Les classes Taquin, jeu2048, et sokoban

correspondent au jeux homonymes, ils héritent de la classe jeu de type <int>.

### 1.3.4 Jeu2048Negative

est une variante du 2048 avec des cases pouvant avoir une valeur négative, deux cases ayant des valeurs opposées s'annulent.

### 1.3.5 Jeu2048TwoThreeFive

est une autre variante du 2048 avec des puissances de 2, 3 et 5.

### 1.3.6 Pour le sokoban(non implémenté)

La classe sokoban héritera de jeux<CaseType>, Enum CaseType : Enumeration représentant les différents types de cases possibles pour le jeu.

## 1.4 Exécution du jeu

### 1.4.1 make

make : compile l'ensemble des fichiers sources (.cpp)

### 1.4.2 ./Jeux

Exécute les jeux en mode terminal (console). A l'exécution de ./Jeux, un menu propose d'exécuter un des jeux. Pour augmenter la taille de la grille, le programme demande à l'utilisateur d'entrer la taille. Un menu apparaît pour le mode d'exécution, le programme demande de choisir le mode joueur où le mode robot.

### 1.4.3 make clean

make clean : sert à nettoyer le répertoire - à supprimer les fichiers .o.