

UNIVERSITÉ D'ANGERS

RAPPORT DE PROJET D'OPTIMISATION  
COMBINATOIRE

---

Placement de bâtiments dans un espace à deux  
dimensions

---

*Présenté par :*

Mateanui ARAPARI  
Divin NDAYISHIMA

Avril 2019

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Choix d'implémentation</b>	<b>3</b>
2.1	Présentation . . . . .	3
2.2	Différentes classes créées . . . . .	3
<b>3</b>	<b>Résultat</b>	<b>5</b>
3.1	Présentation . . . . .	5
3.2	Le rendu visuel final . . . . .	5
<b>4</b>	<b>Mode d'emploi</b>	<b>6</b>
4.1	Utilisation . . . . .	6
<b>5</b>	<b>Conclusion</b>	<b>7</b>

# 1 Introduction

Dans le cadre du cours d'optimisation combinatoire en 1<sup>ère</sup> année de Master Informatique à l'Université d'Angers, il nous est demandé de travailler sur un projet qui consiste à proposer un algorithme glouton pour trouver des solutions à un problème de placement de bâtiments dans un espace à deux dimensions.

Le projet ayant été réalisé en binôme, nous avons eu recours aux outils de version de control tel que la plateforme de dépôt distant **GitHub**.

**Repository du projet Glouton :** <https://github.com/Mateanui/ProjetGlouton>

## 2 Choix d'implémentation

### 2.1 Présentation

Pour réaliser ce projet, nous proposons une méthode d'implémentation qui se porte sur l'utilisation du modèle MVC afin de gérer les interactions entre l'état des algorithmes et l'interface utilisateur.

Le langage Java propose des classes permettant de gérer efficacement les algorithmes demandés. Nous avons donc utilisé ce langage pour répondre aux besoins de ce projet.

### 2.2 Différentes classes créées

Le modèle étant la base de ce projet, implémente toutes les classes permettant de calculer l'emplacement d'un bâtiment sur un terrain. Les classes suivantes représentent le fonctionnement général de l'algorithme glouton :

- Classe Parseur.java :

Cette classe permet de parser un fichier contenant toutes les données nécessaires à la construction d'un bâtiment (nous reviendront plutard sur ce fichier).

Elle peut donc contenir la dimension d'un terrain sur lequel les bâtiments vont être construit, le nombre de bâtiment à créer et toutes les dimensions de chaque bâtiment à construire.

C'est donc dans ce fichier que l'on récupère les données nécessaires pour l'application des algorithmes gloutons.

- Classe Dimension.java :

Cette classe représente tout simplement les dimensions de tout objet à construire.

- Classe AlgorithmeGlouton.java :

C'est dans cette classe que repose le coeur de notre projet. Toutes les méthodes nécessaires au fonctionnement des algorithmes ont été créées dans cette classe.

On peut donc avoir l'initialisation des différents objets comme :

- la dimension du terrain ;

- le terrain étant à deux dimensions, on affecte toutes les valeurs à -1 (un terrain vide, pas de bâtiment) ;

- les positions des bâtiments initialisées à -1 (pas de positions des bâtiments sur le terrain au départ) ;

Les apports visuels en console ou dans une interface graphiques ont été également implémentés dans cette classe. On peut par exemple avoir une affichage d'une grille contenant des chiffres qui corresponds aux indices de chaque bâtiments.

De plus, les algorithmes gloutons ont aussi été implémentés dans cette classe. Ces algorithmes nous permettent

de calculer si un bâtiment peut être placé ou pas sur un terrain donné.

— Les instances des bâtiments :

L'utilisateur aura le choix de choisir les données à traiter. Ce choix est effectué via l'interface graphique que nous verrons plus tard. Ce fichier contient donc toutes les données comme :

- La taille du terrain
- Le nombre de bâtiments
- Et le reste contient la dimension de chaque bâtiment

Concernant la partie visuelle de l'application, nous avons représenté le résultat du calcul sous forme d'une grille contenant des JButton. Chaque bâtiment aura une couleur appropriée.

L'utilisateur a le choix de choisir les bâtiments à construire et il en existe 4 instances de bâtiments (représentées sous forme de fichier texte).

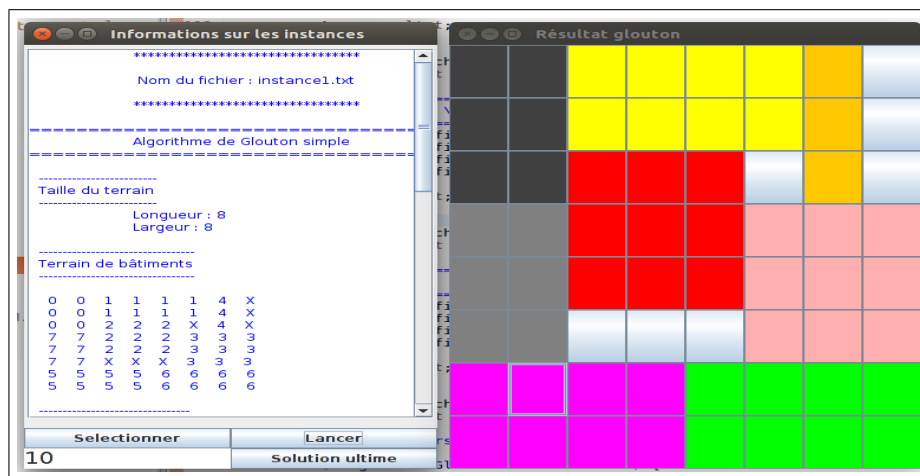
Pour mieux visualiser le travail fourni, nous allons proposer les résultats dans la prochaine partie.

## 3 Résultat

### 3.1 Présentation

Dans cette section, nous allons vous montrer le résultat final de notre projet représenté par des illustrations. Cette partie visuelle est réalisée via la bibliothèque Swing de Javax. Cela nous a permis de présenter un apport visuel et permet d'interagir avec l'utilisateur.

### 3.2 Le rendu visuel final



Cet affichage est le rendu final de notre projet

On distingue deux fenêtres :

— Fenêtre informations :

Le résultat de l'algorithme est affiché sur un support texte illustré à gauche de la fenêtre. Cela nous permet de voir le nom du fichier utilisé, les données utilisées à la construction de cette grille et les positions actuelles des bâtiments.

— Fenêtre resultat :

On remarque sur cette fenêtre la construction final des bâtiments sur un terrain (selon les données fournies)

D'après les données sélectionnées par l'utilisateur, l'affichage se mettra à jour lorsque l'on cliquera sur le bouton "Lancer".

Un mode d'emploi sera présenté à la prochaine section.

## 4 Mode d'emploi

### 4.1 Utilisation

- Exécuter la commande "java -jar nomFileJar.jar"
- Enclenché le bouton "Lancer" pour lancer l'algorithme glouton simple
- Vous pouvez sélectionner une autre instance via le bouton "Sélectionner"
- La saisie de texte qui est par défaut à 10, représente le nombre d'ordres aléatoires pour récupérer la solution optimale
- Pour activer cette algorithme glouton pour la solution optimale, on appuie sur le bouton "Solution ultime", et renvoie donc la meilleur des solutions sur la partie visuel de la grille. Si vous mettez 1000, cela peut prendre plusieurs secondes.

Il va donc falloir scroller dans le text pour avoir plus d'information sur les données utilisées.

Le mode d'emploi sera aussi retranscrit dans le fichier README.md.

## 5 Conclusion

Ce projet qui s'est révélé plus simple que prévu constitue notre première création de programme effectué en optimisation combinatoire. Cela permet de répondre à de plus amples problèmes et la conception de ce projet est plutôt d'un avis positif.

De l'avis général, nous avons renforcé nos connaissances sur l'application de Glouton et nous sommes satisfait du résultat.