

# Snake

Antoine Forgerou

Jérémy Bardon

## Résumé

Dans le cadre du module de logiciel extensible, nous avons développé une plateforme permettant de charger des plugins de manière reflexive. Pour illustrer le fonctionnement de cette plateforme, nous avons recréé le célèbre jeu du snake.

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Plugin Principal</b>	<b>2</b>
<b>3</b>	<b>Plugins Secondaires</b>	<b>4</b>
3.1	L'interface . . . . .	4
3.2	La map . . . . .	4
3.3	Controller le snake . . . . .	4
3.4	Gestion du GameOver . . . . .	4
3.5	Un evenement particulier . . . . .	5

# 1 Introduction

Dans le cadre de notre projet, l'intérêt de développer le jeu du snake est double. Le premier avantage est que le jeu est assez simple à coder, nous permettant de nous concentrer sur la partie "plugins" du projet et non sur le développement du snake en lui-même. Le second est que la plupart des composants du jeu peuvent être des plugins (affichage, map, score, etc ...).

## 2 Plugin Principal

L'application est donc séparée en multiple plugins. Le cœur de l'application est le plugin principale (runnable par la plateforme) appelé **snakecore**. Ce plugin est unique et gère le fonctionnement principale du snake. Pour fonctionner, il utilise des plugins secondaires respectant les besoins qu'il leur impose à travers des interfaces.

Lors du lancement le plugin lance une interface permettant soit de lancer le jeu, soit de lancer une interface de configuration permettant de choisir les plugins utilisées. Des plugins sont choisies par défaut permettant de jouer directement.



FIGURE 1 – Menu du Snake

Dans l'interface de configuration des plugins, les plugins sont triés par type. Pour chaque type, la liste des plugins disponibles est affichés ainsi que le nombre sélectionnable.

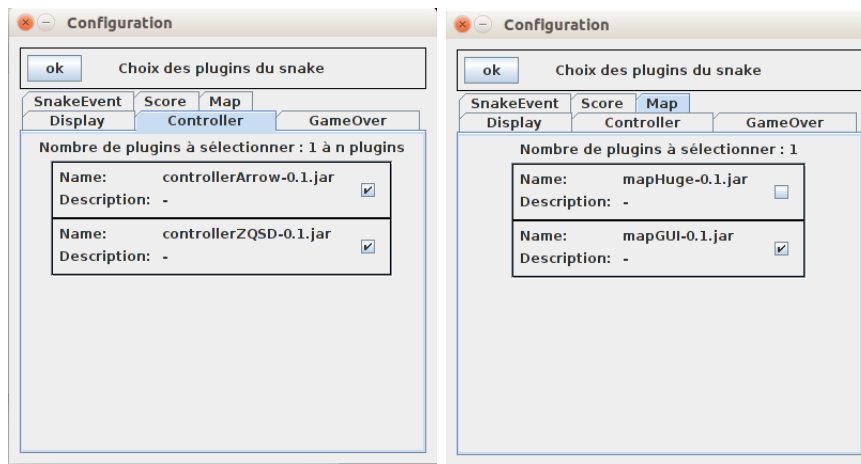


FIGURE 2 – interface de configuration - Controler / Map

Quand le jeu est lancé, le plugin principal demande à la plateforme de charger la liste des plugins sélectionnés afin de pouvoir les utilisées. Une fois les plugins secondaires chargés, le moteur du snake est lancé jusqu'à la défaite du joueur.



FIGURE 3 – Une partie de Snake

## 3 Plugins Secondaires

L'intérêt de ce projet est de séparer chaque fonctionnalité en plugin. Le plugin principal définit les interfaces dont il a besoin ou qu'il supporte. Les plugins secondaires peuvent être classés en plusieurs catégories. En effet certains plugins sont essentiels au bon fonctionnement du programme alors que d'autres sont optionnels. De plus certains plugins peuvent être multiples pour une même catégorie alors que d'autres sont uniques.

### 3.1 L'interface

L'interface d'affichage que va utiliser le plugin principal est un plugin. En effet, il est possible d'imaginer plusieurs interfaces graphiques différentes. Ce type de plugin est nécessaire au fonctionnement du programme. De plus, il est unique car nous souhaitons jouer avec une seule interface.

### 3.2 La map

Ce type de plugin représente la map sur laquelle se déplace le serpent. Le seul intérêt de ce plugin est de spécifier la taille de la map. On pourrait le considérer comme un plugin "data". Ce plugin est aussi nécessaire pour l'application. Pour une partie donnée, il ne peut y avoir qu'une map, ce type de plugin est donc unique.

### 3.3 Contrôler le snake

Pour pouvoir jouer, il est nécessaire de pouvoir changer la direction du snake. Ce contrôleur peut être stocké dans un plugin. Le type Contrôleur est nécessaire au bon fonctionnement, car il est impossible de jouer sans. Nous avons choisi de permettre au joueur de pouvoir jouer avec plusieurs contrôleurs simultanément, cette option n'entrave pas le bon fonctionnement du jeu.

### 3.4 Gestion du GameOver

Ce qui se passe après la défaite du joueur (par défaut contact avec lui-même ou avec un mur) est géré par un type de plugin. Cependant, à travers ce plugin il est possible de modifier les conditions et les comportements de la défaite. Ce type de plugin est nécessaire pour pouvoir gérer la défaite du joueur.

### 3.5 Un evenement particulier

Nous avons souhaité pouvoir ajouter des comportements au cours de la partie. L'intérêt de ce plugin événement est de pouvoir effectuer des actions à plusieurs moments de la partie. Ce type de plugin est optionnel mais peut aussi être multiple.

Exemple :

- Un plugin vitesse → augmente la vitesse de jeu à chaque tour.
- Un plugin evolution → modifie l'agrandissement du snake.