

# Projet Programmation 2

## Première Partie

YOAN GERAN

STEFAN SCHWOON

27 janvier 2023

### 1 Introduction

Le Projet de Programmation 2 consiste cette année en l'implémentation en Scala d'un jeu de stratégie en tour par tour (potentiellement en temps réel par la suite), avec une interface graphique simple.

Dans ce jeu, le joueur contrôle des êtres humains qui se déplacent sur une carte et doivent survivre et se développer en utilisant les ressources présentes sur la carte. Ces dernières permettent notamment de créer des bâtiments aux but variés pour le développement des humains. Cependant, les humains peuvent être attaqués par des ennemis (animaux, autres êtres humains, etc.) ce qui peut mener à la perte de bâtiments voire à la perte du jeu.

Le but du projet est d'utiliser les principes de la programmation orientée objet afin de gérer au mieux les différentes composantes du jeu. Vous travaillerez par groupe de 2 ou de 3, en conservant à l'esprit que nos exigences seront adaptées à la taille du groupe.

### 2 Le jeu à programmer

La première partie du projet consiste à programmer un déplacement d'un personnage (un être humain) sur une carte (qui peut-être fixée à l'avance). Certains endroits de cette carte contiendront une ressource, d'autres contiendront des adversaires qui, une fois battus, rapportent également une ressource.

Des adversaires ou des ressources peuvent apparaître sur la carte lors de certains tours.

#### 2.1 Actions

À chaque tour, on peut choisir de déplacer le personnage sur la carte, de ramasser une ressource ou de frapper un adversaire (si l'on est à côté de la ressource ou de l'ennemi).

#### 2.2 Les adversaires

Si le personnage est suffisamment proche d'un ennemi au début d'un tour, alors l'ennemi le frappe (le personnage a donc un certain nombre de points de vie et la partie est finie s'il les perd tous). Quand un ennemi est vaincu, il laisse tomber une ressource.

## 2.3 Graphique

Pour la première partie il n'est pas demandé d'avoir un design graphique particulièrement esthétique, mais les interactions avec l'utilisateur doivent être intuitives. Les messages au joueur seront affichés par l'interface graphique et non pas dans la console. En particulier, on doit avoir un moyen de connaître son nombre de ressources.

# 3 Évaluation

## 3.1 Rapport et soutenance

Vous devez rendre un rapport de 2 à 3 pages et qui détaille vos choix techniques et les problèmes et difficultés que vous avez rencontrés. Dans le cas où certaines difficultés n'auraient pas pu être surmontées et que votre programme présenterait des défauts, vous pourrez expliquer ici ce que vous avez essayé d'entreprendre pour les résoudre et pourquoi cela n'a pas marché. Une soutenance de 15 minutes par groupe sera organisée à la fin de la première partie du projet où vous nous ferez une démonstration de votre programme.

## 3.2 Fonctionnalité du code

Votre projet sera évalué sur ses fonctionnalités. S'il remplit tout ce qui est demandé, rajouter d'autres fonctionnalités pourront apporter un bonus. La qualité graphique peut jouer un rôle, mais ce cours est avant tout un cours de programmation objet, ainsi, la perspective principale se portera sur les fonctionnalités, et l'évaluation de l'interface graphique reposera avant tout sur son caractère intuitif et facile à prendre en main.

## 3.3 Organisation du code

Votre projet devra être organisé de façon hiérarchique, ou de frapper un ennemi il vous faudra le séparer en fichiers, classes et méthodes. Il vous est recommandé de séparer le plus possible les différentes fonctionnalités, notamment les aspects *interface (frontend)* et *fonctionnement du jeu (backend)*.

## 3.4 Qualité du code

L'évaluation prendra en compte la qualité de votre usage de la programmation orientée objet ainsi que la qualité de votre utilisation du langage Scala. Faites attention à bien utiliser les propriétés d'héritages et à ne pas dupliquer du code inutilement. Utilisez plutôt des directives fonctionnelles que des boucles imbriquées. La mise en forme, la présence de commentaire et la cohérence des noms de classes, méthodes et variables devront être suffisamment décentes pour une lecture agréable du code.

# 4 Dates Importantes

- Deadline pour le rendu du code du projet : mardi 21 février 2022
- Date de la soutenance de la première partie : vendredi 24 février 2022