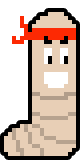
Worms



Rendu Final du Mini Projet

Un projet par : CHARNAY Jacques, BERTRAND Paul, BERNARD Maxime, BOURSAUD Thomas.

Table des matières

[I – Cahier des Charges 1](#_Toc509868080)

[II – Problème posé 1](#_Toc509868081)

[III – Principe de l’algorithme 2](#_Toc509868082)

[IV - Bibliographie 3](#_Toc509868083)

[V – Suggestion d’améliorations du projet, bugs connus 3](#_Toc509868084)

[VI – Carnet de route 3](#_Toc509868085)

I – Cahier des Charges

le cahier des charges de votre programme (quelles fonctionnalités sont proposées à l’utilisateur ?)

Objectif du projet :

Le projet consiste à recréer le jeu “Worms”. Dans ce jeu, plusieurs “worms” s’affrontent sur une carte grâce à des armes variées (lance-roquettes, grenade, fusils, etc ...). Le but est de battre tous les joueurs.

Notre projet sera de créer un terrain entièrement destructible où deux joueurs pourront s’affronter par le biais de leurs “worms”. Les “worms” pourront utiliser 3 armes : lance-roquettes, grenade, fusils. Ces armes détruiront donc le terrain plus ou moins en fonction de leur puissance. La trajectoire des projectiles sera modélisée via un moteur physique.

Notre projet consiste à recréer le jeu « Worms ». Ce jeu est un jeu de type tour par tour en 2D ou des vers de terre, les « Worms », s’affrontent en équipe sur un terrain entouré d’eau. Dans ce jeu, plusieurs Worms s’affrontent sur une carte grâce à des armes variées (lance-roquettes, grenade, fusils, etc ...). Le but est de battre tous les joueurs de l’équipe adverse.

Le but de ce projet est donc de fournir à l’utilisateur une expérience de jeu agréable au plus proche du jeu original « Worms ». Ainsi notre projet remplit les différentes fonctionnalités :

* Générer un terrain praticable pour le joueur et destructible par les armes ;
* Permettre au joueur d’incarner un « Worms » pouvant se déplacer sur ce terrain ;
* Proposer deux types d’armes au joueur, le bazooka et la grenade, pouvant exploser ;
* Les explosions générées par ces armes pourront affecter le terrain et endommager les « Worms » en leur enlevant de la vie ;
* Proposer une expérience de jeu tour par tour similaire à celle des jeux vidéo actuels ;
* Avoir un moteur physique s’appliquant aux « Worms » et aux projectibles.

II – Problème posé

la description du problème posé (quelles sont les problèmes envisagées ?)

Supprimer cette partie ??????????????

III – Principe de l’algorithme

le principe de l’algorithme (comment les problèmes rencontrées sont-ils résolus par le programme)

|  |  |  |
| --- | --- | --- |
| **Classe** | **Classe(s) héritière(s)** | **Description de la classe** |
| Weapons | - Bazooka  - Grenade  - ↳HolyGrenade  *Rmq : HolyGrenade hérite de Grenade* | Classe codant les « lanceurs » de projectiles. Permet de dessiner un indicateur renseignant le joueur sur l’angle du lancer et sur la puissance de celui-ci. |
| Projectile | - Rocket  - GrenadeProjectile  - ↳HolyGrenadeProjectile  *Rmq : HolyGrenadeProjectile hérite de GrenadeProjectile* | Classe permettant l’apparition d’objet mouvant, les projectiles. Cette classe permet de leur appliquer la physique choisie (i.e. la gravité, la puissance du tir). Elle permet également l’explosion de ces projectiles. |
| Menu | ⌀ | Ce menu est une IHM permettant de sélectionner les paramètres préliminaires nécessaires au lancement d’une partie. En effet, ce menu permet de choisir la couleur de son équipe et donc de la couleur de ses Worms ainsi que leurs noms (aléatoirement choisis de base). On peut également choisir le terrain sur lequel on veut s’affronter. |
| FenetreJeu | ⌀ | Classe principale du jeu. |
| MoteurPhysique | ⌀ |  |
| Force | ⌀ |  |
| Worms | ⌀ |  |
| GestionTerrain | ⌀ |  |
| Block | ⌀ |  |
| Inventaire | ⌀ |  |
| GestionTours | ⌀ |  |
| Map | ⌀ |  |

IV - Bibliographie

* *Javadoc de la librairie Slick2D,* [*http://slick.ninjacave.com/javadoc/*](http://slick.ninjacave.com/javadoc/)*:* Cette javadoc nous a grandement aidé pour réaliser notre mini-projet. En effet, nous avons choisi cette librairie afin de pouvoir créer un programme type jeu plus facilement.
* *Tutoriels sur Slick2D et en particulier* [*http://www.shionn.org/tutoriels-slick-2d*](http://www.shionn.org/tutoriels-slick-2d) *:* Ce blog explique comment créer un jeu de type « RPG » grâce à la librairie Slick. Nous avons donc suivi quelques-uns de ces tutoriels afin de réaliser notre projet. Ces tutos nous ont permis de nous familiariser un peu plus avec cette librairie.
* *Les cours du PC :* Les cours du Premier Cycle nous ont grandement aidé pour l’implémentation de l’IHM (le menu) mais aussi pour comprendre les aspects de l’affichage graphique grâce à la librairie Slick. Les listes ont aussi été également utiles.
* *Notre projet sur Github* [*https://github.com/Reddragio/WormsSlick2*](https://github.com/Reddragio/WormsSlick2) *:* Cette plateforme nous a permis de créer une sorte de cloud pour mieux gérer ce projet de groupe.

V – Suggestion d’améliorations du projet, bugs connus

Quelques améliorations du programme peuvent être envisagées. Tout d’abord l’apparition des worms en début de partie qui pourrait être aléatoire. Evidemment, les textures sont perfectibles. Enfin, on pourrait imaginer l’ajout d’armes variées comme des armes au corps à corps. Aussi l’ajout d’items interactifs apparaissant sur la carte pourrait être implémenté.

Idée de bonus:

caisses redonnant de la vie

vent

réglage puissance armes

Animation lors du déplacement

Blocs avec différentes résistances

Les bugs connus sont :

* OUI

VI – Carnet de route

carnet de route et échéancier décrivant l’évolution de votre travail au fil des semaines

un document explicatif de 6 pages maximum au format PDF précisant