

Projet en C

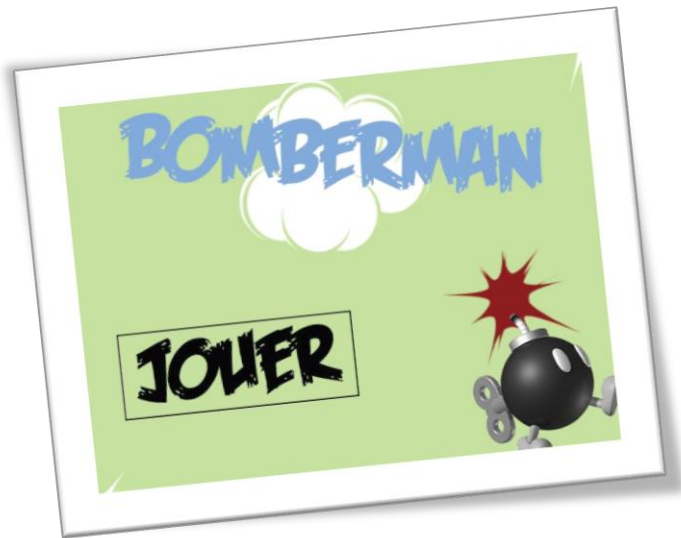
Bomberman

AUTEURS:

- CUNY Kilian
- HABEN Guillaume
- RAME Elise

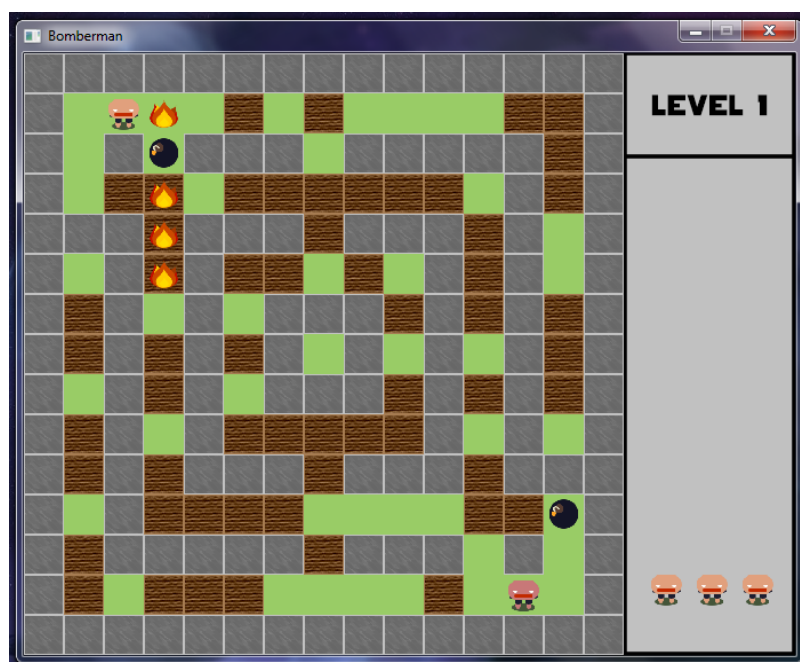
DATE : Avril - Mai 2015

Fiche Bomberman



Le projet que nous avons développé est inspiré du célèbre jeu de réflexion Bomberman dont le principe est le suivant : grâce à des bombes, le joueur doit tuer ses adversaires, des intelligences artificielles, qui ont pour seul but de vous détruire ! Les personnages évoluent dans un environnement rempli de caisses et de murs qu'ils doivent au préalable faire exploser (sans s'exploser soit-même, c'est préférable...) pour atteindre ses ennemis. Le but du jeu est donc de tuer ses ennemis en déposant des bombes sur leur trajectoire afin de réussir à les piéger !

« Snapshot d'une partie »



Sources

Jeu en ligne type Bomberman :

<http://jeu.info/bomberman.htm>

<http://www.jeux.fr/jeux/bomberman>

Une partie des Sprites :

<http://gamedevelopment.tutsplus.com/articles/enjoy-these-totally-free-bomberman-inspired-sprites--gamedev-8541>

Aide pour la structuration de nos classes avec un projet similaire :

<http://makaveli59.free.fr/>

Le projet

Planning prévisionnel du projet

Etape 0 : Git + Virtual Studio

-> Créer un répertoire Git

-> Terminer la configuration des PC sous Visual Studio

Etape 1 : Faire les Tutos C/C++ -> Différence structure / Objet...

Etape 2 : Création structure de base .h/ .cpp (Personnage, Plateau, main...)

Etape 3 : Inclure la SDL2

Etape 4 : Affichage du terrain basique avec les caisses, les murs et le personnage

Etape 5 : Bouger le personnage / Obstacles

Etape 6 : Bombe, mort, vie, Timer, boucle de jeu

Etape 7 : Bonus, plusieurs niveaux

Etape 8 : Musique, Menu, optimisation...

Réalisations

A l'aide de tutoriels en ligne (dont les liens sont disponible dans les sources) nous avons pu inclure la SDL2 à notre projet. Les différentes classes que nous avons utilisées étaient réfléchies au tout départ puis implémentées au fur et à mesure. Ainsi nous avons commencé par créer une classe main, les classe Game et Level permettant de gérer le plateau de jeu, puis les classes Joueur, Bombe et Explosion avant de terminer par la classe Adversaire (implémentation de l'intelligence artificielle). Le menu et le temps ont été créés et modifiés en parallèle tout au long de l'avancé du projet.

La musique a été incluse à la fin du projet, de la même manière que les différentes optimisations de dernière minutes.

Difficultés

2.1) L'intelligence artificielle

Le soucis majeur qui s'est posé à nous aura été celui de l'intelligence artificielle. Nous avons décidé d'implémenter notre propre algorithme récursif de parcours du plus court chemin et nous nous sommes heurtés à beaucoup de cas qui n'étaient pas gérés par notre algorithme ce qui nous a pris beaucoup de temps à corriger.

Une meilleure méthode aurait été de s'intéresser aux différents algorithmes déjà existant comme le célèbre « A Star » qui aurait pu convenir à notre problématique, et essayé de l'adapter à notre projet.

2.2) Organisation du travail et gestion du temps

Une partie non négligeable du temps aura été passée à déboguer notre projet. On a du apprendre à maîtriser cet outil qui s'est révélé indispensable surtout sur Microsoft Visual Studio.

Bien que nous ayons démarré notre projet dès le début du mois d'avril en essayant de répartir notre travail sur toute la plage qui nous était accordée, nous estimons avoir manqué de temps pour finaliser ce projet comme nous aurions voulu le faire. Par ailleurs ce jeu est constamment améliorable et de nombreuses fonctionnalités, que nous avons pensées au départ, comme les bonus ou le jeu contre plusieurs adversaires, n'ont pas pu être développées.