



Projet C++

Bomberman

Koalab koala@epitech.eu

*Abstract: **Bomberman** est l'un des jeux vidéo les plus connus. Avec plus de 70 franchises, depuis l'original sur **MSX**, **ZX Spectrum** et **Sharp MZ-700** en 1983 jusqu'aux dernières versions sur **PlayStation Network**, **WiiWare** et **Xbox Live Arcade** sorties en 2010 ; plus de 10 millions d'unités ont été vendues.*

*Ce projet s'inscrit dans le cadre du partenariat entre **Le Koalab** et le **Game Dev Lab**. Il est donc le fruit d'une collaboration forte entre les **Koalas** et les **Pandas**. J'en profite donc pour remercier sincèrement toute l'équipe du **GDL**, pour leur investissement et la qualité de leur travail, en plus d'être des gens géniaux.*



Table des matières

I	Avant-propos	2
II	Ressources	3
II.1	Bibliothèque	3
II.2	Documentation	3
II.3	Assets	4
II.4	Tutoriel OpenGL	4
III	Partie obligatoire	5
III.1	Avant toute chose !	5
III.2	Généralités	6
III.3	Le jeu	7
III.4	Conseils	8
IV	Consignes générales	9
V	Consignes de rendu	10

Chapitre I

Avant-propos

Nous avons décidé de prendre comme référence le jeu **Neo Bomberman**, sorti sur **Neo Geo** et les systèmes **MVS** en 1997. Il n'est peut-être pas le meilleur parmi toutes les séries mais il a l'avantage d'être conçu spécialement pour jouer à deux joueurs, l'un contre l'autre. Bien sûr, il est possible de jouer contre l'ordinateur.

Eh oui ! Nous voulons que vous créiez un mode multijoueurs permettant de jouer contre les IA dans votre **Bomberman**. L'une des principales différences avec la version **Neo Geo** réside dans le graphisme qui sera en 3D pour les maps et les personnages. Cependant, le gameplay reste en 2D. Toutefois, rien ne vous empêche de tester un gameplay 3D ! Vous pouvez vous inspirer des titres sortis.

- Pour un bref aperçu des titres bomberman : <http://www.uvlist.net/groups/info/bomberman>

Nous vous fournissons une bibliothèque créée par le **GameDevLab** qui contient la base de ce dont vous avez besoin pour charger et afficher des objets 3D. Vous aurez également accès à un pack minimaliste de ressources 3D.

Pour la grande majorité d'entre vous, ça sera votre première expérience dans le développement de jeux vidéo. Vous vous poserez peut-être des questions telles que : comment s'y prendre avec toutes ces fonctionnalités ? Que devrions-nous implémenter en premier ? Etc. C'est pour cela que le **GameDevLab** a rejoint les **Koalas** pour ce projet.

Nous ne vous mentirons pas : la création d'un jeu vidéo n'est généralement pas très amusant (du moins, pas tous le temps). Mais ça peut l'être... et si c'est le cas, à la fin il est plaisant de jouer à un jeu que vous avez développé et de voir les autres s'amuser ! Le résultat mérite des efforts !

Donc, Amusez-vous !

Chapitre II

Ressources

Compte tenu de la nature du projet, le GDL met à votre disposition des contenus pour vous faciliter le développement.

II.1 Bibliothèque

Le GDL a écrit pour votre confort une bibliothèque graphique de qualité basée sur OpenGL. Il va sans dire que l'utilisation de cette bibliothèque est **OBLIGATOIRE** pour ce projet, aucune dérogation possible.

Cette bibliothèque est à votre disposition sur l'intra sur la même page que ce sujet.



Copiez la bibliothèque dans un de vos répertoires locaux, Mais n'oubliez pas de mettre votre copie à jour régulièrement !

Si vous souhaitez utiliser d'autres bibliothèques en supplément, vous DEVEZ soumettre votre proposition à l'équipe du GDL en la justifiant. Nous verrons plus loin comment communiquer avec le GDL.



Votre requête ne sera pas prise en considération si vous n'avez pas implémenté les points minimum du jeu.

II.2 Documentation

Une bonne bibliothèque sans sa documentation est une mauvaise bibliothèque. Vous trouverez donc la documentation de la bibliothèque dans le repertoire doc de l'archive :

1 `libgdl_gl-X.Y/doc/`

II.3 Assets

Même si ce projet comporte une composante graphique très importante, le métier de graphiste reste l'apanage d'une élite. Le GDL met donc à votre disposition des modèles 3D animés pour **Bomberman** et les bombes. Vous DEVREZ toutefois vous débrouiller si vous avez besoin de modèles supplémentaires.

Vous trouverez des assets graphiques mis à votre disposition par le GDL dans le répertoire **assets** de l'archive :

```
1 libgdl_gl-X.Y/assets/
```



Un grand merci à Xavier "Viking" Baures pour son travail sur les modèles !

II.4 Tutoriel OpenGL

La programmation avec **OpenGL** peut se montrer particulièrement cryptique pour un débutant. Afin de mettre toutes les chances de votre côté, le GDL a conçu un tutoriel de qualité pour vous aider à démarrer !

Vous trouverez ce tutoriel au même endroit que ce sujet sur l'intranet **Epitech**. Le lire en entier est particulièrement indispensable.

Chapitre III

Partie obligatoire

III.1 Avant toute chose !

Ce projet est en partenariat avec les Pandas du GDL. Les Pandas n'ayant pas accès au forum C++ de l'intranet Epitech, nous utiliserons le forum du GDL pour communiquer entre étudiants, Pandas et Koalas.

Vous trouverez le forum du GDL à l'adresse suivante : <http://gamedevlab.epitech.net/forum>.

Ce forum contient une section Bomberman qui n'est visible que pour les utilisateurs identifiés sur le forum : <http://gamedevlab.epitech.net/forum/17-bomberman>.

- Vous DEVEZ vous créer un compte sur le forum du GDL.
- Votre nom sur le forum DOIT être votre login Epitech.
- Votre avatar sera automatiquement créé avec votre photo de l'intranet Epitech, vous ne DEVEZ pas mettre d'avatar personnel.
- Vous DEVEZ préciser votre localisation dans votre profil (Paris, Toulouse, Lyon, ...).



Tout compte ne respectant pas ces consignes sera effacé sans notification.



La section Bomberman du forum du GDL sera le SEUL canal de communication officiel pour ce projet, vous DEVEZ l'utiliser. Pandas et Koalas seront actifs pour répondre au mieux à toutes vos interrogations.

III.2 Généralités

Vous DEVEZ coder un Bomberman-like fonctionnel, principalement inspiré par le titre "Neo Bomberman" :

- Un jeu complet comprenant au moins :
 - Intro
 - Menu, modes, options, pause, ...
 - Enregistrer et charger une partie
 - Victoire et défaite
 - Scores persistants et classement
- 2 joueurs sur le même clavier.

Comme vous êtes sur PC, avec les processeurs et la puissance graphique que nous avons, nous ajoutons quelques défis obligatoires :

- un gameplay 2D comme la version originale, mais un graphisme 3D est obligatoire avec des animations et de bons FX.
- Un nombre de joueurs illimité, dont 1 OU 2 contrôlés par des joueurs humains. Les autres joueurs seront contrôlés par l'ordinateur.
- La possibilité de choisir parmi des maps prédéfinies par vos soins ET parmi des maps générées aléatoirement. Il ne DOIT PAS y avoir de taille maximum pour une map. C'est à dire qu'on DOIT pouvoir générer une map de 100 cases par 100 cases avec 10 joueurs sans ralentissement.
- La perte de performance ne sera tolérée que pour de gigantesques maps avec des centaines de joueurs.



FIGURE III.1 – Neo Bomberman in game

III.3 Le jeu

- Le client DOIT afficher une vue du haut des cases avec les murs et les boîtes... C'est la map.
- Les joueurs DOIVENT être déplaçables par les touches fléchées (ou l'équivalence pour le second joueur sur un clavier unique), pouvoir poser une/des bombe(s) et entrer en collision avec les obstacles (murs, boîtes, bombes).
- Les bombes DOIVENT exploser dans les quatres directions après un certain laps de temps, avec une certaine portée et détruire le premier élément destructible rencontré.
- La vitesse des mouvements et des animations DOIT être indépendante du matériel informatique.
- Chaque joueur sur une même map DOIT avoir un skin différent (en terme de couleur). Ce point nécessite de la réflexion de votre part.
- Si les deux joueurs humains sont trop loin l'un de l'autre pour apparaître sur le même écran, vous DEVEZ trouver une solution qui permet de continuer à jouer normalement.
- Au début de la partie, chaque joueur ne peut poser qu'une seule bombe. Cette bombe explose en couvrant 2 cases dans les 4 directions.
- Il DOIT y avoir des bonus dont au moins les 3 suivants :
 - Pouvoir poser une bombe supplémentaire.
 - Augmenter la portée des bombes d'une case.

- Augmenter la vitesse de déplacement du joueur de 1 unité.
- Chaque ordinateur (IA) DOIT être scripté. Proposer plusieurs scripts peut rendre une partie plus amusante.
- Les points d'apparition DOIVENT être dispersés de manière aléatoire ET les plus espacés possible.
- Les scores DOIVENT être enregistrés et affichés sur le tableau des scores qui DOIT perdurer d'une exécution du jeu à une autre.
- Les parties DOIVENT être enregistrables et chargeables sans limite. Une date et un petit screen shot de la partie seraient très appréciables pour les identifier facilement.
- Votre Bomberman DEVRAIT être multi-threadé à l'aide de la bibliothèque `pthread`.
- Votre Bomberman DEVRAIT avoir de la musique et des effets sonores.

Vous êtes complètement libres pour ce qui est du script des IAs. Vous pouvez même utiliser des langages tels que LUA pour les réaliser si vous le souhaitez.

De manière similaire, vous êtes libres d'utiliser ce qui vous chante pour sérialiser/dé-sérialiser les données dont vous avez besoin pour sauvegarder/charger des parties ou des scores.

III.4 Conseils

Pensez simple et efficace. C'est un peu simple comme conseil mais vous devez le garder en tête. Si cela se complexifie, faites un retour en arrière, réfléchissez de nouveau et posez des questions si besoin ! Il n'y a rien de complexe dans le sujet, mais si voulez le réussir, vous devez comprendre comment les différentes parties du jeu interagissent entre elles.



L'intranet Epitech vous laisse créer des groupes de 4 à 6 personnes. Toutefois, faire des groupes de 6 personnes peut s'avérer être une bonne stratégie compte tenu du travail à fournir.

Chapitre IV

Consignes générales

Vous êtes globalement libres de faire l'implémentation que vous voulez. Cependant, il y a quelques restrictions :

- Il est **OBLIGATOIRE** d'avoir un compte sur le forum du GDL.
- Si vous constatez une anomalie technique sur ce forum, merci d'en avertir les auteurs de ce sujet au plus vite s'il vous plaît.
- L'utilisation de la bibliothèque fournie par le GDL est **OBLIGATOIRE**.
- Les seules et uniques fonctions de la `libc` autorisées sont celles qui encapsulent les appels système et qui n'ont pas d'équivalent en C++.
- Toute bibliothèque qui n'est pas explicitement autorisée est explicitement interdite, dans les limites du contexte du ce sujet.
- Toute réponse à une problématique **DOIT** être une approche objet.
- Toute valeur passée par copie plutôt que par référence ou par pointeur doit être justifiée, sinon vous perdrez des points.
- Toute valeur non `const` passée en paramètre doit être justifiée, sinon vous perdrez des points.
- Toute fonction membre ou méthode ne modifiant pas l'instance courante mais n'étant pas `const` doit être justifiée, sinon vous perdrez des points.
- Il n'existe pas de norme en C++. Cependant, tout code que nous jugerons illisible ou trop sale pourra être sanctionné arbitrairement. Soyez sérieux !
- Tout branchement supérieur à `(if ... else if ... else ...)` est **INTERDIT** ! Factorisez !
- Gardez un œil sur ce sujet régulièrement car il est susceptible d'être modifié.
- Nous attachons une grande importance à la qualité de nos sujets, donc si vous trouvez des fautes de frappe, d'orthographe ou des incohérences, merci de nous contacter à l'adresse koala@epitech.eu pour que nous puissions y remédier dans la journée.

Chapitre V

Consignes de rendu

Vous DEVEZ rendre votre projet sur le dépôt fourni par Epitech.
Le nom de dépôt pour ce projet sera `cpp_bomberman`.



Assurez-vous que votre dépôt a correctement été créé et que les droits dessus sont corrects. Aucune excuse ne sera tolérée en soutenance.

Vos dépôts seront ramassés à l'heure exacte de la fin du projet, intranet Epitech faisant foi. Venir se plaindre que "sur ma montre il était pas 23h42" ne sert à rien. La seule heure valide à nos yeux est celle d'Epitech car le système est automatique.

Corollaire à loi de Murphy : "Si tu rends ton travail dans la dernière heure, quelque chose va mal se passer".

Seul le code présent sur votre dépôt sera évalué lors de la soutenance.

Bon courage !

