



So Long

Et merci pour le poisson !

Résumé:

*Ce projet est un jeu 2D simple conçu pour vous faire utiliser des textures,
des sprites et quelques éléments basiques de gameplay.*

Version: 1

Table des matières

I	Introduction	2
II	Objectifs	3
III	Règles communes	4
IV	Partie obligatoire	5
IV.1	Le jeu	6
IV.2	Gestion graphique	6
IV.3	La carte	7
V	Partie bonus	8
VI	Exemples	9
VII	Rendu et peer-evaluation	10

Chapitre I

Introduction

Être dev, c'est super pour créer son propre jeu.

Cependant, un bon jeu nécessite de bonnes ressources. Afin de créer des jeux 2D, vous devrez rechercher des *tiles*, *tilesets*, des *sprites* et des *sprite sheets*.

Fort heureusement, des artistes de talent partagent leur travail sur des plateformes telles que :

itch.io

Bien entendu, veuillez respecter le travail d'autrui.

Chapitre II

Objectifs

Il est temps pour vous d'attaquer votre premier projet graphique !

so long vous donnera des bases dans les compétences suivantes : gestion de fenêtre, gestion des événements, choix de couleurs et de textures.

Vous allez maintenant prendre en main la bibliothèque graphique de l'école : la **MiniLibX** ! Cette bibliothèque a été développée en interne et inclut des outils basiques permettant d'ouvrir une fenêtre, de créer des images et de gérer des événements clavier et souris.

Les objectifs de ce projet sont similaires à tous ceux de votre première année : faire preuve de rigueur, vous améliorer en programmation C, utiliser des algorithmes basiques, chercher des informations en autonomie, etc. ...

Chapitre III

Règles communes

- Votre projet doit être écrit en C.
- Votre projet doit être codé à la Norme. Si vous avez des fichiers ou fonctions bonus, celles-ci seront incluses dans la vérification de la norme et vous aurez 0 au projet en cas de faute de norme.
- Vos fonctions de doivent pas s'arrêter de manière inattendue (segmentation fault, bus error, double free, etc) mis à part dans le cas d'un comportement indéfini. Si cela arrive, votre projet sera considéré non fonctionnel et vous aurez 0 au projet.
- Toute mémoire allouée sur la heap doit être libéré lorsque c'est nécessaire. Aucun leak ne sera toléré.
- Si le projet le demande, vous devez rendre un Makefile qui compilera vos sources pour créer la sortie demandée, en utilisant les flags `-Wall`, `-Wextra` et `-Werror`. Votre Makefile ne doit pas relink.
- Si le projet demande un Makefile, votre Makefile doit au minimum contenir les règles `$(NAME)`, `all`, `clean`, `fclean` et `re`.
- Pour rendre des bonus, vous devez inclure une règle `bonus` à votre Makefile qui ajoutera les divers headers, librairies ou fonctions qui ne sont pas autorisées dans la partie principale du projet. Les bonus doivent être dans un fichier différent : `_bonus.{c/h}`. L'évaluation de la partie obligatoire et de la partie bonus sont faites séparément.
- Si le projet autorise votre `libft`, vous devez copier ses sources et son Makefile associé dans un dossier `libft` contenu à la racine. Le Makefile de votre projet doit compiler la librairie à l'aide de son Makefile, puis compiler le projet.
- Nous vous recommandons de créer des programmes de test pour votre projet, bien que ce travail **ne sera pas rendu ni noté**. Cela vous donnera une chance de tester facilement votre travail ainsi que celui de vos pairs.
- Vous devez rendre votre travail sur le git qui vous est assigné. Seul le travail déposé sur git sera évalué. Si Deepthought doit corriger votre travail, cela sera fait à la fin des peer-evaluations. Si une erreur se produit pendant l'évaluation Deepthought, celle-ci s'arrête.

Chapitre IV

Partie obligatoire

Nom du programme	so_long
Fichiers de rendu	Makefile, *.h, *.c, quelques cartes
Makefile	NAME, all, clean, fclean, re
Arguments	Une carte au format *.ber
Fonctions externes autorisées	<ul style="list-style-type: none">• open, close, read, write, malloc, free, perror, strerror, exit• Toutes les fonctions de la bibliothèque mathématique (option de compilation -lm, man man 3 math)• Toutes les fonctions de la MiniLibX• ft_printf et tout équivalent que VOUS avez codé
Libft autorisée	Oui
Description	Vous devez créer un jeu 2D basique dans lequel un dauphin s'échappe de la planète Terre après avoir mangé du poisson. Au lieu d'un dauphin, de poisson et de la Terre, vous pouvez utiliser le personnage, les items et le décor de votre choix.

Votre projet doit respecter les règles suivantes :

- Vous **devez** utiliser la MiniLibX. Soit la version disponible sur les machines de l'école, soit en l'installant par les sources.
- Vous devez rendre un **Makefile** qui compilera vos fichiers sources. Il ne doit pas relink.
- Votre programme doit prendre en paramètre un fichier de carte se terminant par l'extension **.ber**.