

# PROJET

## MY\_BIRD

2023



Bienvenue à tous pour cette activité de découverte du langage C et de la bibliothèque graphique CSFML à travers la création d'un petit jeu graphique nommé "my\_bird".

Pendant cette activité, vous aurez la chance de plonger dans le monde de la programmation en C et d'explorer les fonctionnalités offertes par la bibliothèque CSFML pour créer votre propre jeu.

Vous découvrirez les bases de la programmation en C, notamment la syntaxe, les structures de données et les fonctions, ainsi que les concepts clés liés à la programmation de jeux, tels que la boucle de jeu et la gestion des événements.

Au fur et à mesure que vous progresserez dans cette activité, vous serez en mesure de mettre en pratique ces concepts et d'appliquer vos nouvelles compétences à la création de votre propre jeu graphique "my\_bird".

Nous espérons que cette expérience vous permettra non seulement de vous amuser en créant votre propre jeu, mais aussi de vous initier à la programmation en C et de vous donner une base solide pour vos futurs projets de développement.

Prêt à vous lancer dans la programmation de jeux en C avec la CSFML ? Alors, c'est parti !

## ETAPE 1

- Téléchargez les fichiers sur github
- Ouvrir le fichier : src/main.c

### ■ TIPS

En C, toutes les instructions finissent par “;” comme en français, à la fin de chaque phrase, on met un “.”.

## ETAPE 2

- Tous les commentaires commencent par "//" :

```
static float x = 0.0f; // Position actuelle du sprite
```

- Les zones à remplir sont encadrées par "/\*" et "\*/" :

(Ne pas toucher à ceux qui sont marqués « ETAPE »)

```
sfRenderWindow_drawSprite(/* Fenêtre */, /* Sprite */, NULL);
```

- Vous pouvez commencer à remplacer par le bon code les zones encadrées.

### ■ TIPS

N'hésitez pas à vous aider d'internet et des dernières technologies ainsi que du document `doc_strucutre.md` où vous pourrez y retrouver quelques aides très précieuses !

De plus, n'hésitez pas à regarder ce qui a été écrit précédemment 😊

## ETAPE 3

Et si maintenant on venait modifier notre code pour que notre jeu fasse plus de choses ?

Ce que vous pourriez faire maintenant, c'est que quand le joueur gagne, le jeu affiche un message de fin avec le nombre de passages que le sprite a fait (voir ligne commentaire `/* ETAPE 3 */`)

Exemple :

Bravo, tu as gagné en 4 tours !

### ■ TIPS

Allez voir la fonction "printf" elle fait des choses marrantes et utiles

## ETAPE 4

Est-ce qu'on peut modifier la vitesse/difficulté du jeu ? Pas encore !

Ce qu'on pourrait faire à présent, c'est que ton jeu puisse prendre un nombre en entrée pour la difficulté (plus le nombre est grand, plus c'est difficile).

Pour lancer le programme, il faudrait maintenant rajouter un nombre qui serait la difficulté : `./my_bird 124`

Vous pouvez remplacer :

```
17     sfRenderWindow_setFramerateLimit(my_game->window, 120);  
18     /* ETAPE 4 */
```

Par :

```
if (nb_arg == /* 12 % 5 = ? */) {  
    my_game->speed_sprite = /* fonction qui permet de passer un char * et int */(tab_arg[1]);  
} else {  
    my_game->speed_sprite = /* Vitesse si jamais l'on ne donne pas de vitesse */;  
}  
sfRenderWindow_setFramerateLimit(my_game->window, my_game->speed_sprite);
```

## ■ TIPS

Pour la première zone, regardez ce que fait l'opérateur modulo (= "%") sur internet, vous pouvez utiliser une calculatrice.

## ETAPE 5

Pas mal, pas mal

Je trouve que la difficulté n'est pas assez élevée, notre oiseau commence toujours au même endroit. Et si désormais, on le faisait commencer d'une position aléatoire ?

Ajoutez cette ligne :

```
74     /* ETAPE 5 */  
75     my_game->y = (rand() % (550 - 50 + 1)) + 50;
```

Cette partie de code nous permet de venir donner une position aléatoire en y de notre sprite. La fonction `rand()` donne un nombre aléatoire et le calcul `"% (550 - 50 + 1) + 50"` permet de modifier ce nombre pour obtenir un nombre aléatoire entre 50 et 550.

## ETAPE 6

Maintenant que tu as presque fini ton jeu, on pourrait venir un peu le customiser ?

- Ce que tu pourrais faire c'est modifier le titre de la fenêtre :

## ETAPE 7

Tu pourrais aussi modifier le fond du jeu :

- TIPS

Regardez dans le dossier “ressource” et regardez ce qui se passe dans la fonction “init\_structure” 😊

## ETAPE 8

Ce qu'on peut ajouter maintenant, c'est que lorsque le jeu est ouvert, on puisse le fermer aussi grâce à une touche si l'on appuie dessus. (exemple : ECHAP)

```
/* ETAPE 8 */
```

- TIPS

Regardez ce que fait la fonction "sfKeyboard\_isKeyPressed" 😊

Inspire-toi de ce qui a été fait précédemment dans le code.



Et voilà, nous arrivons à la fin de notre activité de création de jeu. J'espère que vous avez apprécié cette expérience de développement et de création de votre propre jeu.

Mais cela ne s'arrête pas là ! Il y a encore beaucoup de potentiel d'amélioration et de fonctionnalités que vous pouvez ajouter à votre jeu pour le rendre encore plus intéressant et amusant :

- Pour commencer, vous pourriez ajouter des boutons supplémentaires pour offrir plus d'options et de fonctionnalités aux joueurs. Par exemple, vous pourriez ajouter un bouton pour passer au niveau suivant ou pour accéder à un menu de pause.
- Un menu principal et un menu de fin avec un score enregistré pourraient également être une excellente addition. Cela permettrait aux joueurs de voir leur score et leur progression, ainsi que de revenir en arrière et de rejouer le jeu plusieurs fois pour améliorer leur score.
- Si vous êtes à la recherche d'un défi supplémentaire, vous pouvez également ajouter un joueur IA pour jouer contre les joueurs humains. Cela ajouterait un élément de compétition et d'excitation au jeu.

En fin de compte, les possibilités sont infinies, et vous êtes le seul à décider de la direction que vous souhaitez prendre pour améliorer votre jeu. Nous espérons que cette activité vous a inspiré à continuer, à créer et à développer vos propres jeux à l'avenir. Merci d'avoir participé à cette expérience de création de jeu et bonne continuation !

Si jamais vous avez des questions n'hésitez pas à contacter [tragris#4539](#) sur discord 🗨