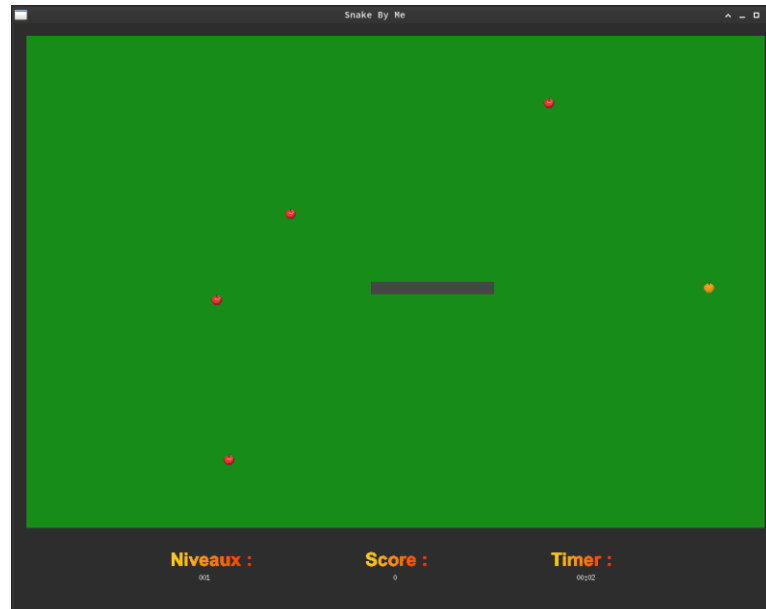


1. Introduction :

Le projet consiste en la création d'un jeu Snake (Serpent) en utilisant le langage de programmation C et la bibliothèque graphique de l'université. L'objectif principal est de mettre en œuvre les fonctionnalités classiques d'un jeu Snake, en ajoutant une fonctionnalité plus personnelle à l'aide de la programmation en C et avec pour but de nous laisser explorer divers moyens de programmation pour atteindre un objectif.



2. Fonctionnalité :

Le principe et le fonctionnement du jeu Snake est plutôt simple et ne nécessite pas une grande quantité de fonctionnalité :

- Initialiser () :

La première des fonctionnalités et de d'initialiser les jeux pour son bon fonctionnement avec la fonction Initialiser () qui renvoie une structure que nous avons créer pour simplifier le transfert de donnees par exemple comme attribut d'une fonction. Nous initialisons le nombre de ligne et de colonne ainsi que la position du serpent ainsi que sa direction.

- InitialiserSnake () :

Qui dessine l'affichage du jeu avec le background de la fenêtre ainsi que le terrain de jeux et dessine le serpent initial de taille 10. Puis charge l'image de la barre d'information.



Niveaux :

Score :

Timer :

- `checkDeplacement ()` :

La fonction `check déplacement` sert à vérifier si le déplacement qui a été effectuer mettra fin a la partie ou non.

- `deplacerSnake ()` :

Cette fonction gère le déplacement du serpent en fonction de la direction et en sauvegardent l'emplacement ou était la queue pour pouvoir l'effacer ultérieurement.

- `majSnake (InfosJeux, Snake)` :

Cette fonction met à jour l'affichage du serpent si nécessaire en dessinant un carré de la couleur du terrain de jeux sur la case qui été la queue du serpent avant son déplacement et en dessinant la tête du serpent.

- `majText ()` :

Cette fonction met à jour le texte écrit dans la barre d'information plus précisément le niveau qui est en rapport avec la taille du serpent, le score et pour finir le timer.

- `gameOver ()` :

Cette fonction gère l'affichage de l'écran de fin de partie qui affiche le score.



- `gameStart ()` :

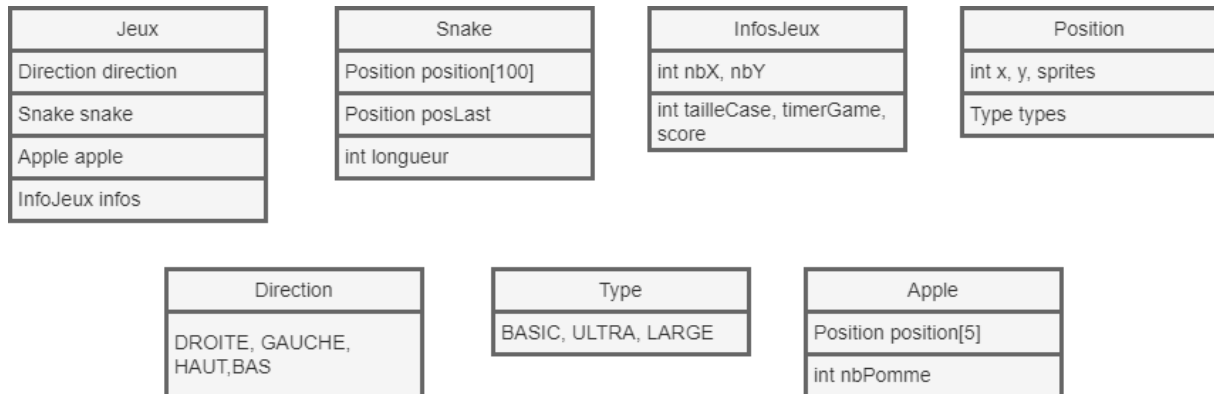
Cette dernière fonction gère le fonctionnement du Snake en appelant les fonctions aux moment adéquats ainsi que la génération des différentes pomme.



2. Découpages et Explication des fonctions :

J'ai divisé mon code en une multitude de structure et d'énumération pour simplifier l'accès aux informations ainsi que la modification.

L'une des principales structures est le Jeux qui contient toutes les données nécessaires aux bons fonctionnements du jeu :

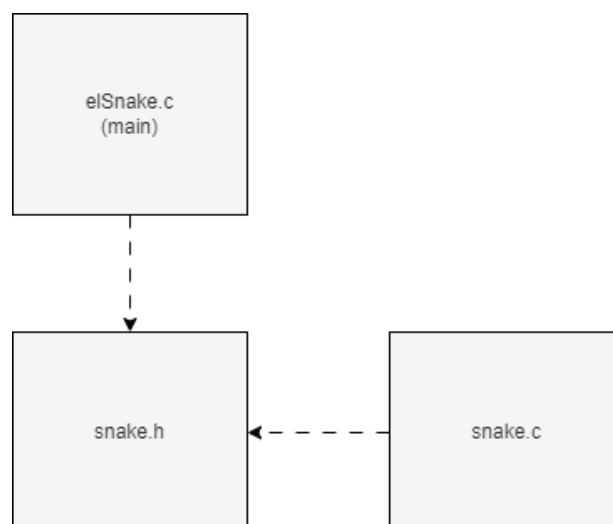


Après nous avons la structure Snake qui possède un tableau de position qui est une autre structure qui possède 4 attributs x, y, sprites ainsi que types un héritage de la structure Type qui est une énumération, reprenons avec snake et son deuxième attribut posLast qui la l'ancienne position de la queue du serpent et pour finir longueur qui est la taille du serpent.

Nous avons aussi infosJeux qui possède comme attributs nbX, nbY respectivement le nombre de colonne et de ligne qui ne sont pas destinés à être changés comme tailleCase qui est évidemment la taille d'une case ensuite nous avons timerGame qui est le temps en seconde passé depuis le début de la partie qui évolue en continu et le score qui est explicite et qui évolue évidemment.

Ainsi que Apple qui possède un tableau de 5 positions qui évoluera à chaque fois qu'une pomme est mangée et donc régénérée ainsi que nbPomme qui ne changera pas.

Voici la répartition des fichiers utilisés pour le makefile.



3. Conclusion Personnel :

Moi Bourama-Mike Djoco j'ai trouvé que ce projet m'a permis de me remettre dans une optique de projet, réfléchir avant de coder car aux débuts de ce projet j'ai immédiatement codé mais après réflexion ils y avaient beaucoup d'erreur j'ai donc pris la décision de reprendre depuis le début. Et de travailler en C avec la bibliothèque graphique que je n'ai pas très utiliser l'an passé et j'ai bien aimé le sujet et j'ai pu apprendre de mes erreurs.

Et Ibrahima Bah , personnellement, j'ai trouvé ce projet très enrichissant. La gestion des entrées utilisateur, la manipulation des sprites, et la gestion des collisions ont été des aspects particulièrement intéressants. J'ai également apprécié la collaboration avec mes coéquipiers pour résoudre les problèmes et atteindre nos objectifs communs