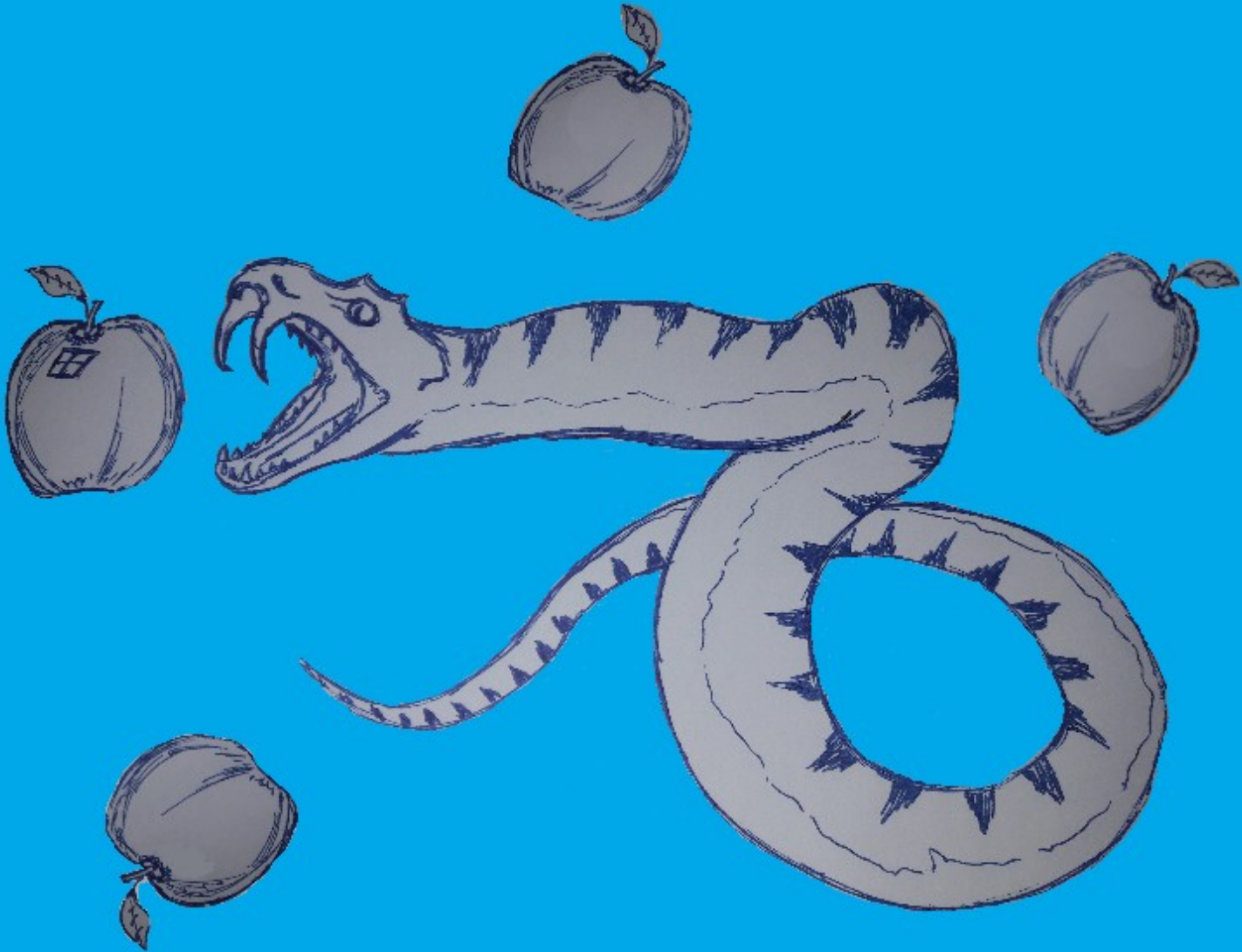


PT 1.1 APL 2019/2020 :

Jeu du Serpent



SOMMAIRE

I/ Introduction au sujet

.....

II/ La structure du programme

.....

III/ Explication de la forme des données qui représentent le serpent dans le programme, et des transformations que ces valeurs peuvent subir au cours de la partie

.....

IV/Conclusions personnelles.

.....

I/ Introduction au sujet :

Le but de notre projet est de réaliser une version du jeu du Snake en C89. Dans un terrain rectangulaire, le joueur contrôle un serpent toujours en mouvement qui doit manger des pommes sans rentrer dans sa propre queue, percuter les limites du terrain. Mais attention ! Plus le joueur va faire manger de pommes au serpent, plus celui-ci va grandir rendant la tâche plus ardue pour chaque point gagné.

En somme, il est quasiment impossible de gagner pour un utilisateur lambda.

Notre programme possédera les fonctionnalités suivantes :

- Le jeu affichera le temps (en minutes et en secondes) écoulées depuis le début de la partie.
- La possibilité de mettre la partie en pause en appuyant sur la touche espace du clavier.
- Un système de score dynamique.

Pour rendre les parties de notre jeu plus pimentées, nous avons rajoutés quelques fonctionnalités supplémentaires :

- L'ajout de pommes de couleurs différentes possédant toutes un effet différent des autres.

-Pommes dorées (Pommes donnant plus de score que les normales de couleur rouge).

-Pommes argentées (Pommes pouvant donner des effets aléatoires comme le fait de quadrupler la vitesse du serpent pendant une courte période de temps, de doubler la longueur du serpent progressivement, de faire en sorte que tous les points obtenus par le joueur seront doublés pour le reste de sa partie ou encore rien du tout avec de la chance. Il s'agit d'une pomme piège en quelque sorte).

-Pommes violettes (Pommes qui doubler le score pendant 30 secondes, l'effet se cumule mais la durée reste la même. On peut donc arriver à des bonus de score extravagants si l'on réussit à en obtenir rapidement un certain nombre à la suite).

II/ La structure du programme

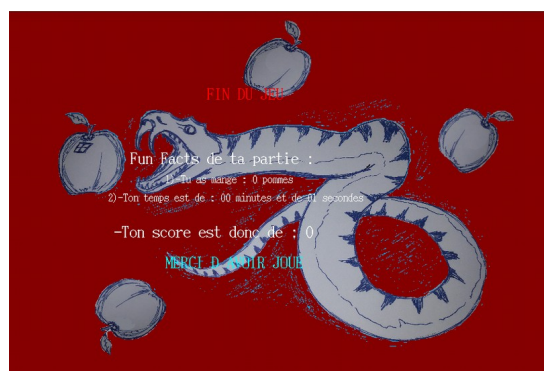
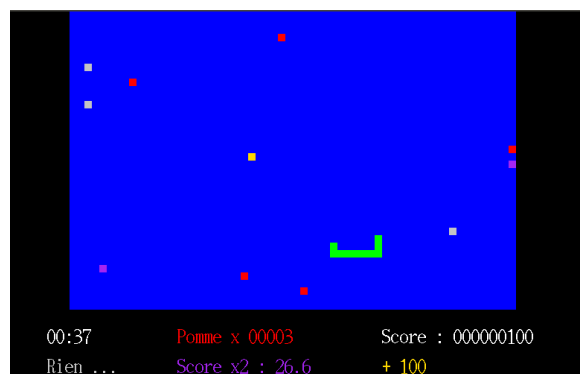
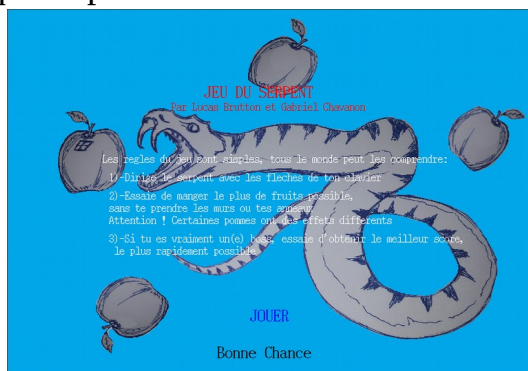
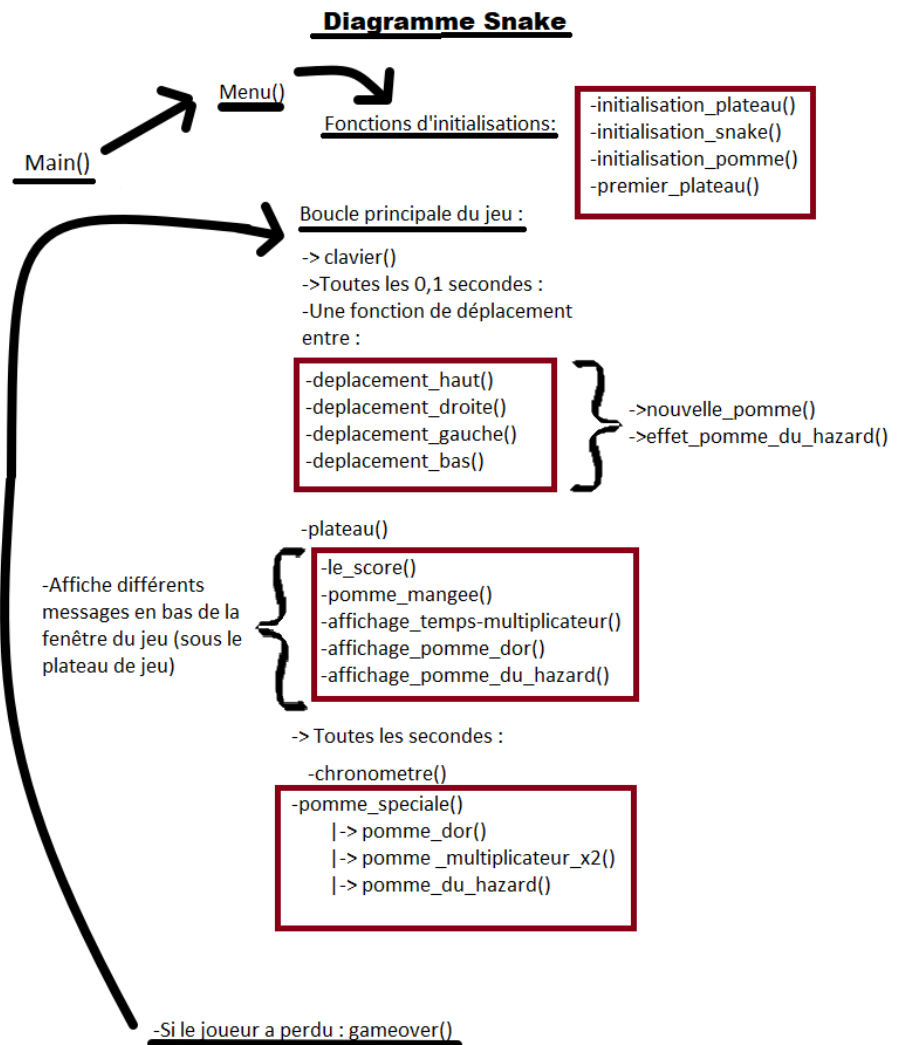
Le programme est composé de plusieurs fonctions qui vont s'appeler entre elles.

Tout d'abord, quand le programme est lancé, la page du menu s'affiche, en cliquant avec la souris, la page se ferme et le plateau est initialisé, la partie commence immédiatement, dès que le joueur perd, la fenêtre du plateau se ferme à son tour et une dernière fenêtre est ouverte, celle de la fin de partie. Le programme peut être donc être quitté de nouveau par un clic de la souris.

Comme montré dans le diagramme ci-contre, le main va d'abord appeler le menu qui va ensuite activer les fonctions d'initialisations.

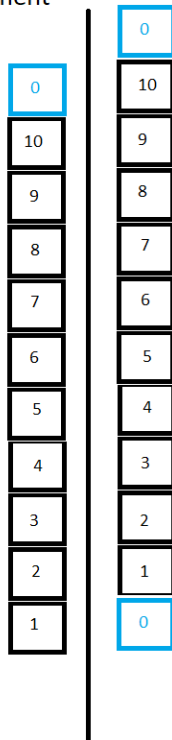
Ensuite, on entre dans la boucle principale du programme :

Cette boucle ne s'arrête pas tant que la touche espace ou échap n'est appuyée ou bien que la partie se termine.



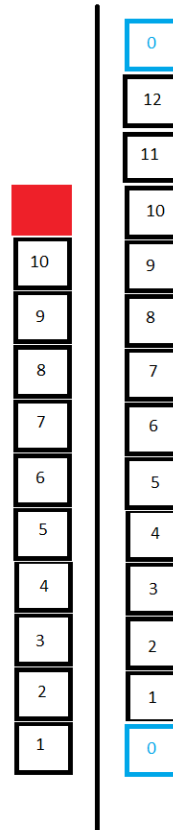
III/ Explication de la forme des données qui représentent le serpent dans le programme, et des transformations que ces valeurs peuvent subir au cours de la partie

1) Déplacement



-> -1 pour toutes les cases quand le serpent se déplace sur le plateau

2) Manger une pomme



-> Quand le serpent mange une pomme, la tête qui est rentrée en contact avec va devenir le nouveau bout de la queue et ne va recommencer à bouger avec le reste du corps seulement quand elle arrivera au bout de celui-ci

Les valeurs du serpent changent constamment dans le programme.

Le principe du déplacement du serpent est le suivant :

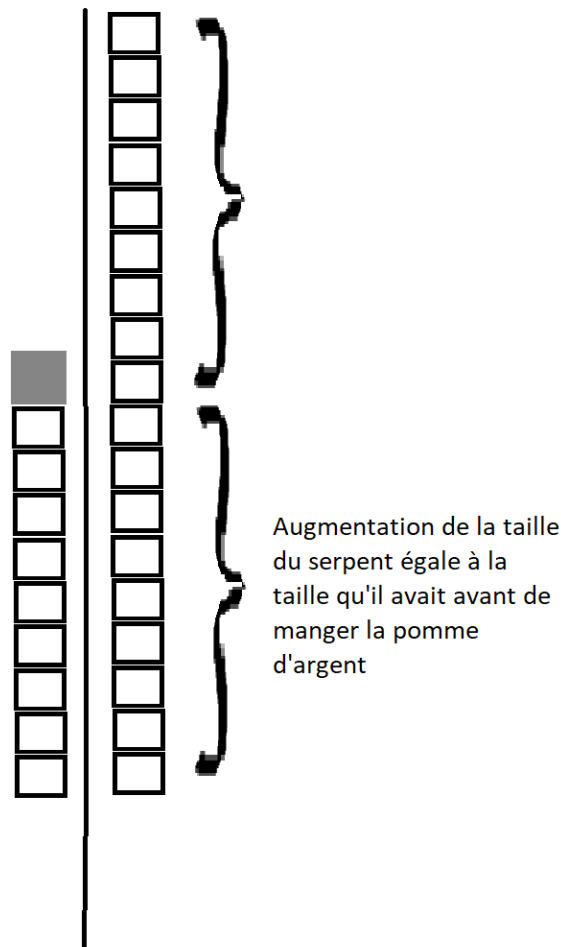
Le serpent est au début de la partie, composé de 10 cases. La tête du serpent est numérotée 10, la suivante est numérotée 9 et ainsi de suite jusqu'à 1.

Quand la tête va avancer d'une case, la valeur de la tête actuelle passe à 9, la suivante à 8 jusqu'à ce que la valeur de l'ancien bout de la queue devienne 0 (valeur des cases vides du plateau) donc, cette case change de couleur et fait donc partie du plateau. La case où le serpent avance devient sa tête provisoire et donc le processus se répète. Quand le serpent mange une pomme la tête gagne +2 pour sa valeur.

Le principe du doublement de la longueur du serpent via les pommes aléatoires :

La variable qui représente la longueur du serpent qui est multiplié par 2.

Le système fonctionne comme pour les pommes classiques (voir schéma), mais au lieu d'augmenter la taille du serpent de 2, la longueur du serpent augmente de la taille précédente du serpent qu'il avait avant de manger la pomme.



IV/Conclusions personnelles

Conclusion personnelle (Gabriel Chavanon)

Ma participation au projet :

Les fichiers qui j'ai personnellement codés sont :

- menu.c
- gameover.c
- Dessin de l'image de fond de l'écran de menu et de fin de partie.
- J'ai aussi apporté mon aide comme j'ai pu pour le reste du programme.

Les difficultés rencontrées :

Les difficultés que j'ai rencontrées au cours du projet étaient :

- D'écrire des fonctions et de prendre en compte ce qu'elles font faire dans le programme
- D'utiliser des pointeurs à travers multiples fonctions;
- De lier la partie graphique et code du jeu;
- Et pour terminer, de trouver comment représenter le Snake.

Conclusion :

J'ai personnellement trouvé le sujet du projet assez intéressant car il m'a permis d'avoir un regard nouveau sur l'utilité des tableaux dans la programmation et qu'il s'agit de mon premier jeu/projet en équipe en C. Cette réalisation de projet a été très instructive car elle m'a permis d'approfondir mes connaissances, car le projet m'y a poussé. J'ai par ailleurs appris certaines « bêtises » à ne pas reproduire pour le prochain projet. D'ailleurs, bien que notre projet soit terminé nous aurions pu y rajouter quelques améliorations. Par exemple l'ajout d'un système de classement en fonction des scores les plus hauts qui serait estimé en fonction du temps et des points des joueurs pour créer une certaine forme de compétitivité, il pourrait y avoir des classements pour chaque catégories (temps, points, etc). Pour conclure, ce projet a été très amusant et j'ai vraiment apprécié le réaliser en équipe.

Conclusion personnelle (Lucas Bruton)

Les fichiers codés :

Les fichiers que j'ai personnellement codés sont :

- main.c
- pomme.c
- serpent.c
- plateau.c

Les difficultés rencontrées :

Les difficultés que j'ai rencontrées au cours du projet étaient :

- D'écrire des fonctions qui prennent en entrée des tableaux à deux dimensions;
- D'utiliser des pointeurs à travers multiples fonctions;
- D'organiser l'effacement de mon code;
- De lier la partie graphique et code du jeu;
- De trouver certaines erreurs ;
- De coder un chronomètre;
- Et pour terminer, de trouver comment représenter le Snake.

Conclusion :

J'ai trouvé que le sujet du projet était très bien choisi puisqu'il était amusant, et qu'il permettait de voir l'évolution du programme à chaque nouvelle fonction rajoutée. De plus, coder le Snake a été une très bonne expérience éducative. En effet, le projet m'a permis d'approfondir ma compréhension de certaines notions du C dues aux nombreuses difficultés rencontrées, mais également de connaître les difficultés rencontrées lorsqu'on travaille en équipe tel que divisé le travail. En effet, Gabriel voulait s'occuper des fonctions qui créaient les pommes. Cependant, après avoir terminé ma partie, j'avais une idée de comment implémenter les pommes à mon code que j'ai tout de suite rajouté. En faisant cela, on peut dire que j'ai « volé » une partie du travail de mon binôme.

Pour terminer, s'il y a un point que j'aurais aimé améliorer dans ma partie, ce serait le côté graphique du jeu qui est très basique, puisque je n'ai utilisé que des carrés. Mais je suis tout de même satisfait du résultat que Gabriel et moi avons produit.