# RAPPORT DE PROJET TUTORÉS N°1

## Le Snake


LEOTURE Romain
RAYMOND Enveric
IS1/grp2



Année universitaire 2019-2020

# Table Des Matières

# Introduction

Dans le cadre de notre première d'année à l'Institut Universitaire de Technologie de Sénart-Fontainebleau nous avons eu pour tâche la réalisation d'un projet tuteurés en informatique. Notre objectif était de réaliser un snake en utilisant le langage de programmation C89 ainsi que la bibliothèque graphique de l'IUT.

# Partie I : Analyse du jeu

## A- Présentation générale

Le Snake, autrement dit le serpent en français est un jeu créé à la fin des années 70. Ce jeu consiste à diriger un serpent afin de lui faire manger des pommes qui apparaissent de façon aléatoire sur le terrain de jeu, chaque pomme mangée rapporte des points et fait grandir le serpent, le but de ce jeu est d'avoir le plus de points ainsi que le plus grand serpent possible. Évidemment il y a des obstacles sans cela le jeu serait trop simple, les obstacles présents sont les bords du terrain et le corps du serpent, entrer en contact avec ces obstacles termine la partie. De plus nous voulions créer un Snake original, ce dernier est un dragon chinois qui vole au milieu des immeubles d'une ville, les pommes sont remplacées par des oiseaux.

## B- Déroulement du jeu

Une fois le programme lancé le serpent apparaît au milieu du terrain, il se déplace seul vers la gauche et en continu il ne vous reste plus qu'à prendre en main le Snake à l'aide des flèches directionnelles. N'ayez crainte si vous devez vous absentez un moment au bon milieu d'une partie il est possible de mettre le jeu en pause en appuyant sur la touche « espace » de votre clavier vous pouvez relancer le jeu à tout moment en réappuyant sur la même touche. Si vous souhaiter quitter le jeu appuyer simplement sur la touche « échap ». À la fin d'une partie vous pouvez consultez votre score et votre temps.

# Partie II : Gestion de projet

## A- Cahier des charges

Des règles supplémentaires sont à respecter. Notre terrain de jeu prend la forme d'une grille de 40 lignes par 60 colonnes. Le serpent doit apparaître au milieu du terrain, sa taille initiale est de 10 segments (1 segment équivaut à 20 pixels). Les pommes sont aux nombres de 5 pastilles, elles font la taille d'une case (20 pixels), elles réapparaissent lorsqu'elles sont mangées et le serpent s'allonge de 2 segments lorsqu'il en consomme une, une pomme consommée augmente le score de 5 points.

Le joueur peut changer la direction du serpent à l'aide des touches directionnelles ← ↑ → ↓. Il peut quitter la partie avec la touche Esc, et mettre le jeu en pause avec la touche Espace (il met fin à la pause en rappuyant sur Espace).

En dehors du terrain de jeu le joueur peut voir le temps écoulé et le score actuel. Ces informations sont visibles à la fin de la partie sur la fenêtre « Game Over » afin que le joueur puisse juger sa performance.

Les variantes que nous avons intégrées dans notre projet sont des obstacles fixes que le serpent doit éviter. Notre but étant de représenter un Snake sous la forme d'un dragon dans les airs nos obstacles sont des immeubles qui sont présent sur la bordure du bas de notre terrain (cette bordure représente le sol).

## B- Répartitions des tâches

Pour optimiser notre temps de travail nous avons divisés le projet en plusieurs parties :

Enveric étant en charge de la création de la fenêtre graphique, des obstacles (bordures du terrains, immeubles) de la gestion de la nourriture, du « mode Game Over », du score et du Makefile.

Romain étant en charge de la gestion du serpent, (ses déplacements, son agrandissement…). Du chronomètre, de l'initialisation du mode pour mettre la partie en pause ou bien pour y mettre fin.

# C- Description de la réalisation

## C.1- Création d'une fenêtre graphique

Notre premier objectif a été bien entendu de créer la fenêtre graphique. Notre choix a été de la créer en 1200/880 pixels ; une première « partie » de la fenêtre (le terrain de jeu) qui fait donc 1200/800 pixels pour obtenir des cases de 20 pixels par 20 pixels afin d'atteindre la demande de 40 lignes par 60 colonnes, sur cette fenêtre graphique nous avons « posé » un fond de couleur bleu nuit en adéquation avec l'atmosphère de notre jeu. (Un dragon chinois dévorant des oiseaux la nuit en milieu urbain). Les 80 pixels restants seront utilisés comme zone d'affichage (encart) du chronomètre et du score. Nous avons aussi créé un affichage si le joueur perd que nous avons mis dans la fonction gameover.

## C.2- Apparition du dragon chinois

Dans un second temps, nous avons commencé à créer le dragon. Vu la difficulté de la tâche, au début notre dragon a été représenté par une seule case afin de tester ses déplacements. Notre démarche a été d'avancer minutieusement et d'être logique pour résoudre les bugs au fur et à mesure. Ce qui nous a permis d'avoir un raisonnement logique, une fois ce raisonnement en tête nous avons créé la structure du serpent et son corps.

## C.3- Création de la nourriture

Par la suite, nous avions pour objectif de faire apparaître aléatoirement 5 pastilles qui correspondaient à la nourriture, une fois que nous avions réussi cette objectif notre but était de les faire apparaître seulement sur le terrain de jeu, une fois ce dernier terminer, il ne nous restait plus qu'à les faire réapparaître une fois que le serpent les avait mangés.

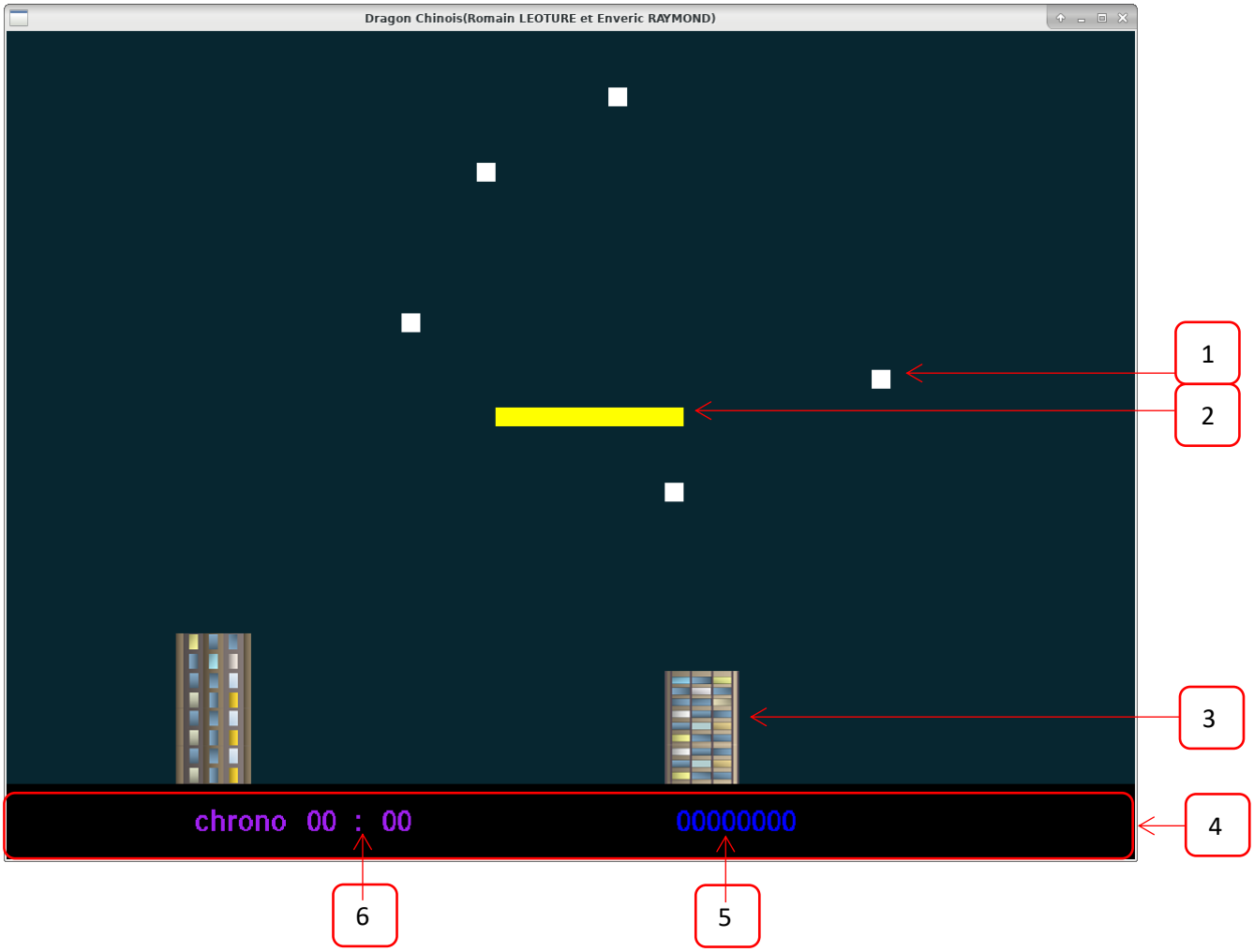
## C.4- Environnement du jeu

Enfin, un fois le dragon créé et la nourriture fonctionnelle nous nous sommes attelés à créer l'environnement du jeu et les obstacles. Nous avons instauré les bordures du terrain de jeu, les bordures des immeubles et les collisions avec son propre corps, nous avons fait en sorte que lorsqu'il y avait une collision dragon contre les bordures cela entrainerait un game over. Pour finir, il ne nous restait plus qu'à mettre en place le chronomètre, le score, la possibilité de mettre le jeu en pause et de pouvoir quitter ce dernier grâce à la touche échap.

# Partie III : Fonctionnalités du programme

## A.1- Interface graphique

Au lancement du jeu, le Snake d'une taille de 10 segments apparaît au milieu de l'écran, les pastilles de nourriture apparaissent à des endroits différents.




1 ***Nourriture*** (oiseau), la nourriture est déterminée à l'aide du tableau « oiseaux[5] » et apparaît aléatoirement sur le terrain de jeu grâce à une fonction « RAND ».

2 ***Dragon*** (Snake), la taille du dragon est initialisée à l'aide de la variable « longueur_dragon » qui est initialiser à 10, sa position de départ quant à elle est déterminé par le tableau « POSITION dragon[COLONNE*LIGNE] ».

3 ***Immeuble*** (obstacle), Afin qu'ils soient perçus comme des obstacles nous avons créé une condition qui fait appel à la fonction « gameover » seulement si les coordonnées de la tête du dragon sont égales aux coordonnées qui délimite la zone des immeubles.

4 ***Encart***, pour créer cet encart nous avons utilisé la fonction « RemplirRectangle. »

5 ***Score***, il est géré à l'aide d'un type char « char score[100] » et de la variable « scor » Cette variable et ce type sont initialisés à zéro au début de la partie. Le score est affiché dans l'encart à l'aide de la fonction « sprintf ».

6 ***Timer***, Le Timer est défini par un « char minutes[100] », un « char secondes[100] », des unsigned long « sec », « min » tout deux définies à 0 et un « temps ». Le « temps » est égale à la fonction « Microsecondes » plus le « CYCLE » qui est définit à 1000000L Le Timer est affiché dans l'encart à l'aide de la fonction « sprintf ».

## A.2- Interface graphique

Quelques temps après son lancement nous pouvons voir que le jeu à évolué, le temps s'est écoulé, le score a augmenté et le dragon est devenu plus grand car il a mangé, la nourriture est donc réapparue à des endroits différents.

1 ***Nourriture***, à l'aide d'une boucle « for » nous pouvons déterminer si un oiseau a été mangé, si le nombre d'oiseaux présent sur le terrain est inférieur à « NOURRITURE_MAX » (c'est-à-dire 5) alors on incrémente de 1 la valeur du tableau oiseaux pour faire réapparaitre un oiseau afin qu'il y ai constamment 5 oiseaux sur le terrain.

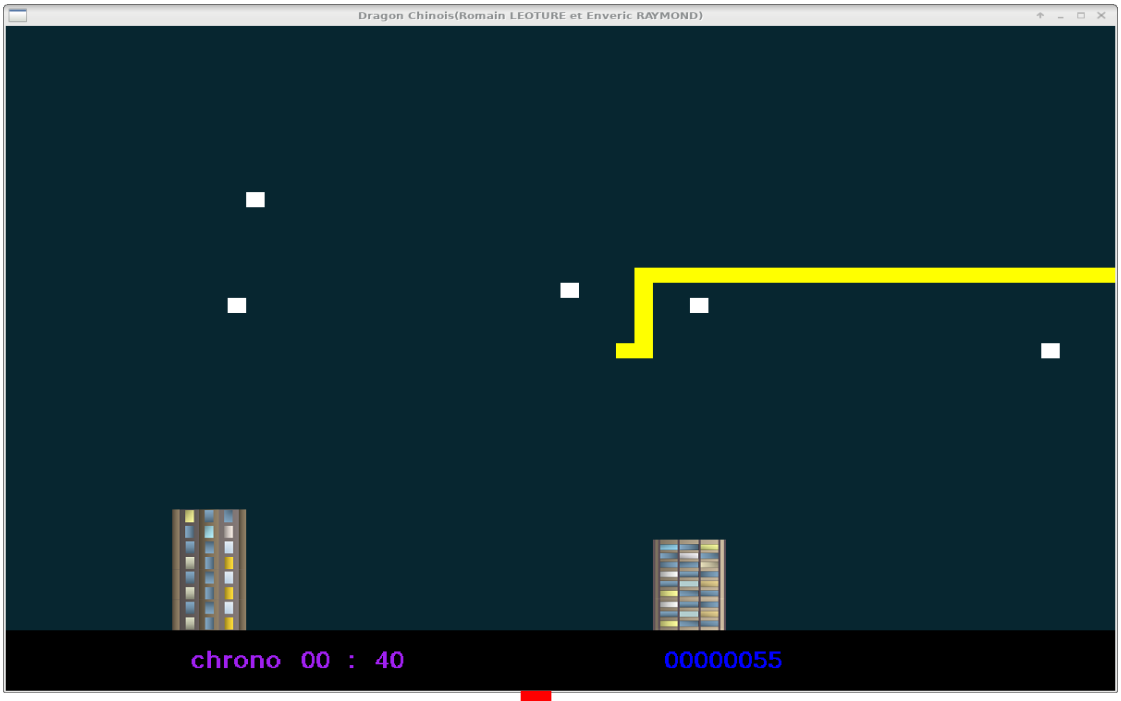
2 ***Dragon***, l'utilisation d'une boucle « if » qui incrémente de 2 la valeur de la variable « longueur_dragon » seulement si les coordonnées de la tête du dragon sont égales à celles d'un oiseau (donc que le dragon a mangé un oiseau) permet d'augmenter la taille du dragon.

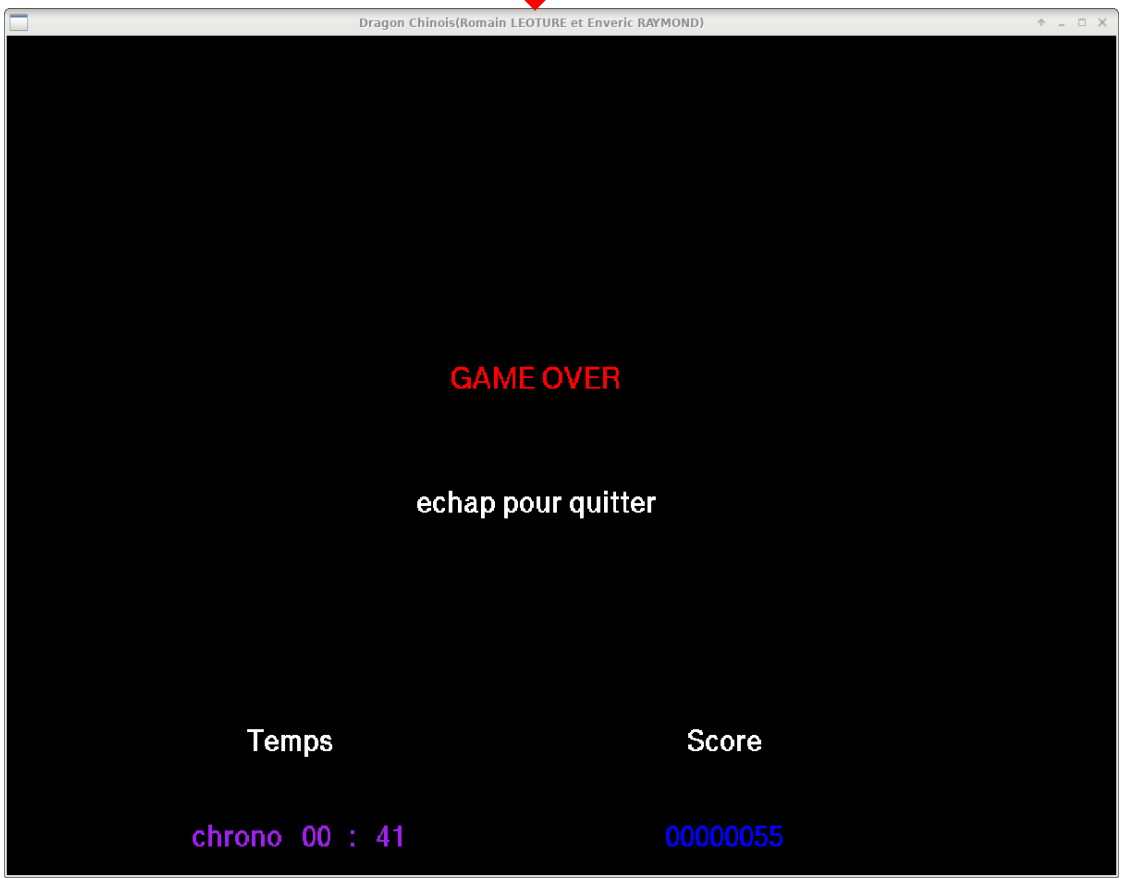
3 ***Score***, à chaque fois que « longueur_dragon » augmente (c'est-à-dire à chaque fois que le dragon mange) la variable « scor » est incrémentée de 5.

4 ***Timer***, *grâce à un « if » nous pouvons déterminer si « Microsecondes » est supérieur à « temps » si cela est le cas on incrémente de 1 « sec » (c'est-à-dire que toutes les secondes on augmente le temps d'une seconde).*

## A.3- Interface graphique

Lorsque le dragon rentre en contact avec un obstacle, la fenêtre GAME OVER s'affiche.



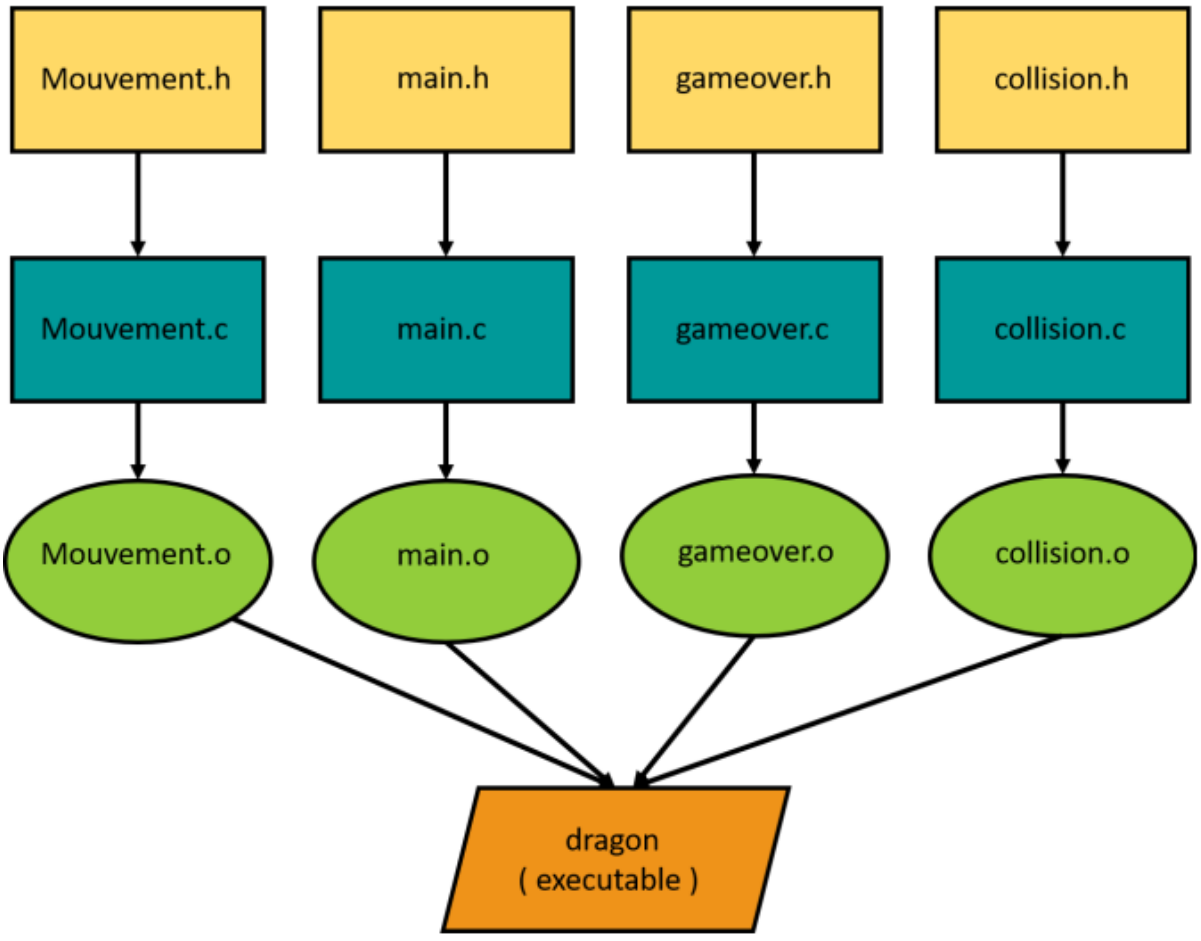



L’affichage de cette fenêtre est rendue possible grâce à un appel à la fonction « gameover ». Cet appel intervient lorsque la tête du dragon rentre en contact avec un obstacle (c’est-à-dire lorsque « dragon[0] » a les mêmes coordonnées que « dragon[i] » ou que « dragon[0] » a des coordonnées supérieurs aux bordures du terrain ou que « dragon[0] » a les mêmes coordonnées que les surfaces des bâtiments).

# Partie IV : Structure du programme

Afin de rendre agréable la compréhension de notre code nous avons organisé des fonctions dans des fichiers à part (sous-fichiers), puis nous avons effectué un Makefile sur l’ensemble de ces fichiers. De plus cela simplifie l’exécution du programme.

Diagramme de la structure du programme



# Partie V : Conclusion

## A- Bilan personnalisé de Enveric

Ce projet s'est avéré extrêmement enrichissant. Déjà en terminal j'ai travaillé sur l'amélioration d'un ancien jeu conçu en 1987 et qui s'intitulait « 1942 ». C'était un projet réalisé en python.

Grâce à ce nouveau projet, « le dragon chinois », j'ai pu améliorer ma compréhension globale de la création d'un programme car dans ce sujet il s'agissait de partir d'une « page blanche » et pas seulement d'améliorer un jeu. Il s'agissait donc d'un véritable travail de création qui m'a permis de plus de mettre en pratique l'utilisation du langage C.

Par ailleurs, le travail d'équipe s'est avéré très positif et indispensable car chacun a pu apporter ses connaissances et ses idées. Ce travail d'équipe très formateur m'a donné confiance en mes capacités même si je réalise que j'ai encore beaucoup de connaissances à assimiler. Dans la programmation je pense que le travail de groupe est fondamental pour avancer et ne pas rester bloqué sur une difficulté. Ainsi, dans notre projet afin de ne pas perdre de temps, nous avons décidé d'un commun accord de limiter notre dragon sur une seule case nous laissant ainsi le temps de la réflexion pour résoudre la difficulté du mouvement du dragon.

Mes regrets concernant ce projet sont liés à la gestion du temps. J'aurai souhaité avoir davantage de temps pour apporter plus de fonctionnalités au jeu. De plus, afin d'obtenir un jeu plus complet, j'aurai aimé mettre en place plusieurs niveaux. Il s'agit donc pour moi lors d'un nouveau projet de prendre en compte cette gestion du temps et des difficultés qui peuvent survenir à tout moment

## B- Bilan personnalisé de Romain

Pour ma part ce projet est une réussite car nous avons réussi à accomplir toutes les instructions demandées. J'ai rencontré de multiples difficultés tout au long de ce projet, en causes mes lacunes en programmation, comme des problèmes de lisibilités de mon code, d'organisation, la plus grosse difficulté que j'ai rencontrée était la création de la structure du Snake. Ce projet fut très instructif, la réalisation de ce Snake m'as permis d'approfondir mes connaissances du langage C89, d'améliorer ma gestion de projet au travers de tâches multiples et variées ainsi que de pallier mes difficultés.

Étant un ancien élève de STI2D, je suis habitué à la réalisation de projet en groupe, ces connaissances m'ont permis d'être en adéquation avec mon camarade. Ce projet m'a fait prendre conscience une nouvelle fois de l'importance et de la nécessité d'un travail d'équipe performant pour optimiser mon temps de travail.

J'ai particulièrement apprécié ce projet, en plus d'être un enseignement ludique je trouvais le concept très intéressant.