

Projet du Serpent

Table des matières

3-4 Présentation du projet

5-7 Présentation des fonctionnalités du programme

8-13 Le Makefile

14-15 Les changement du serpent

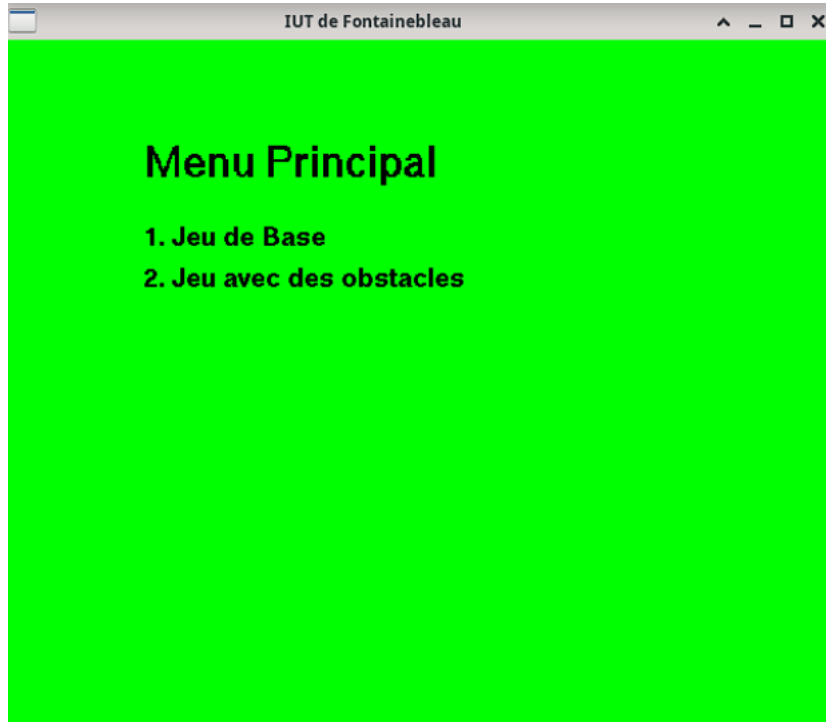
16-17 Conclusions des auteurs

1. Présentation du projet

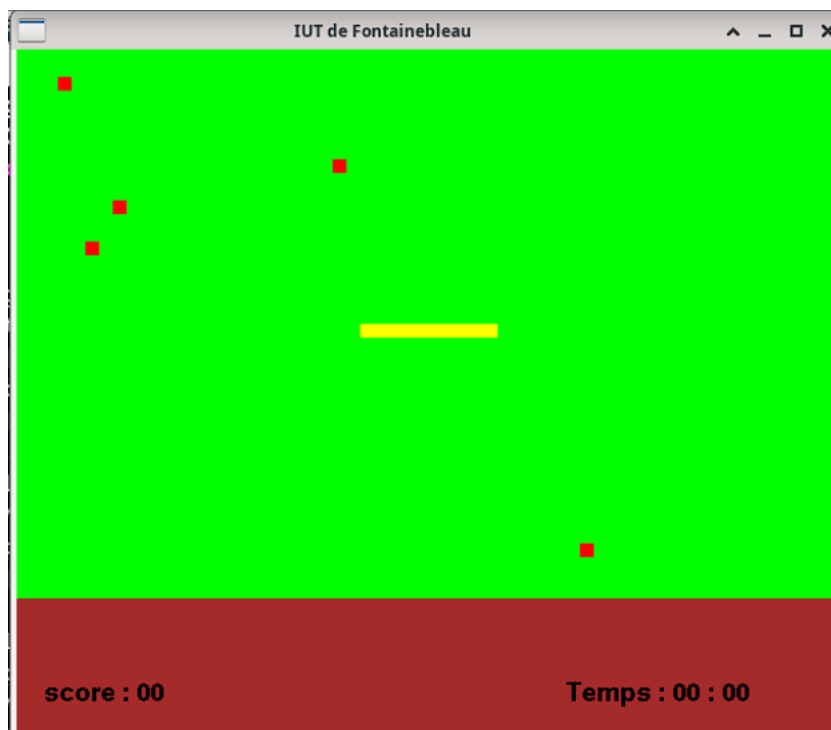
Ce projet est un jeu réalisé en C89 avec la bibliothèque graphique de l'IUT de Fontainebleau où avec les flèches directionnelles du clavier, on peut choisir la direction du serpent, la touche espace doit servir pour mettre le jeu en pause, et la touche échap doit fermer le programme. Le serpent doit manger des pastilles rouges pour grandir et augmenter le score du joueur, il y a aussi le temps de jeu du joueur affiché à l'écran. Ce projet doit fonctionner avec un Makefile, dont avec la commande « make » on compile les fichiers et la commande « make run » pour lancer le programme. Dans le projet, il doit y avoir deux modes de jeu, un où il n'y a que le serpent et un autre avec une variante, pour nous, ce sont des obstacles bleus qui se déplacent à intervalle défini. Si le serpent touche un obstacle où qu'il sort du terrain le joueur perd la partie. Lorsque le joueur perd la partie plus rien ne bouge pendant quelques instants ce qui permet au joueur de voir son temps et son score.

2. Présentation des fonctionnalités du programme

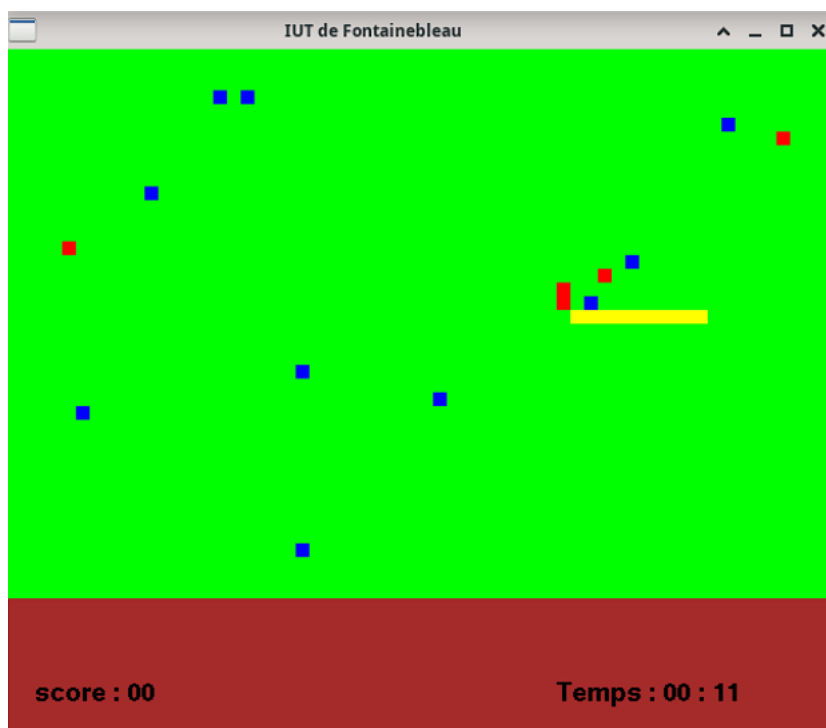
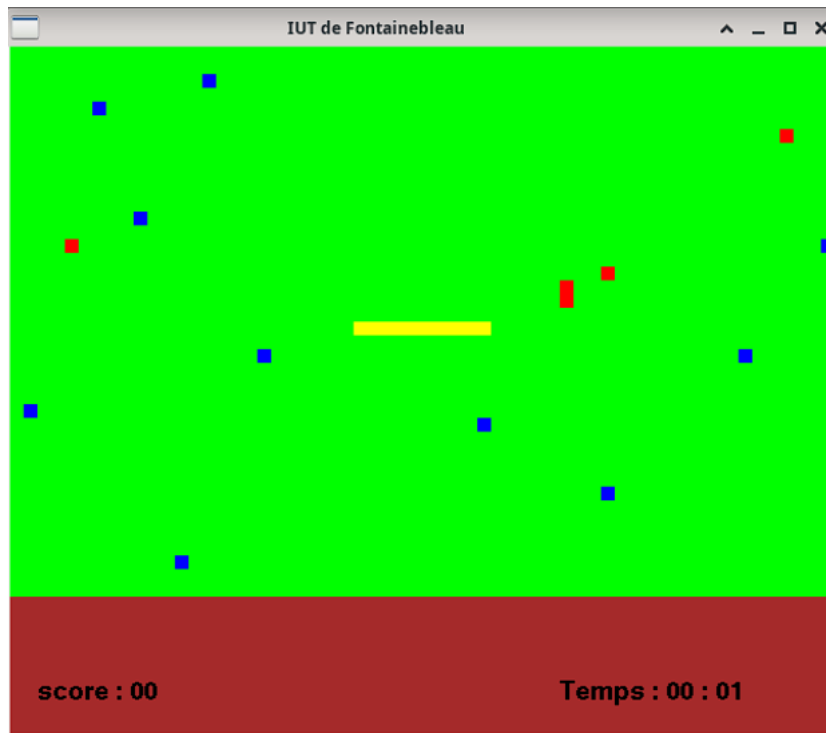
Au lancement du programme, on peut choisir un des deux modes de jeu :



Si l'on choisit le jeu de base, on arrive dans le jeu où il faut manger les pastilles rouges :

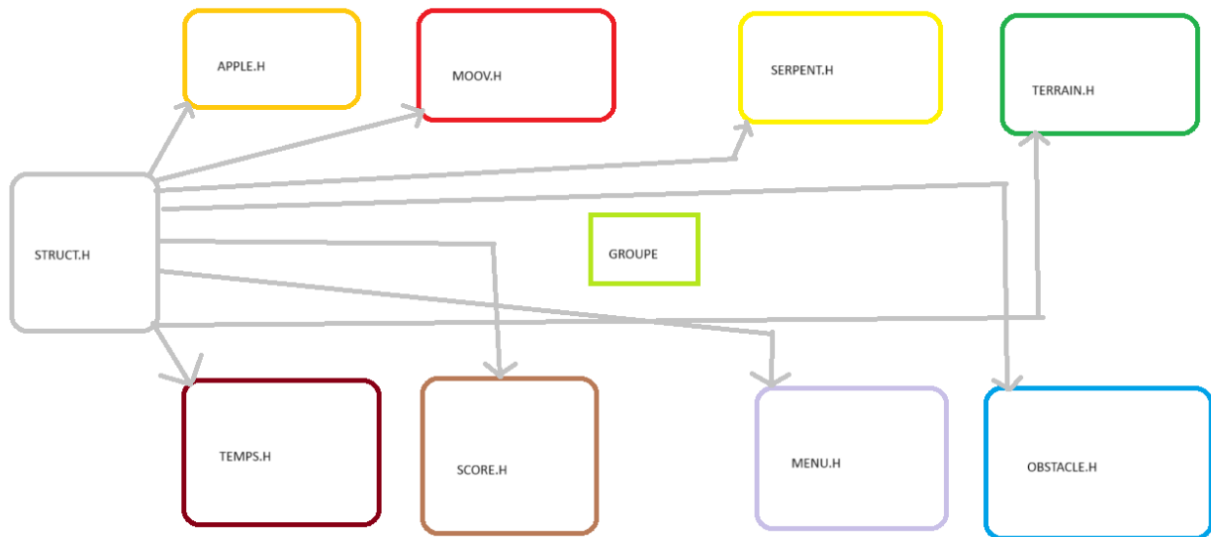


Si le joueur choisit l'autre mode de jeu, il se retrouve sur le même terrain avec les mêmes pastilles, mais cette fois-ci avec des obstacles bleu qui après un certain moment se déplace :

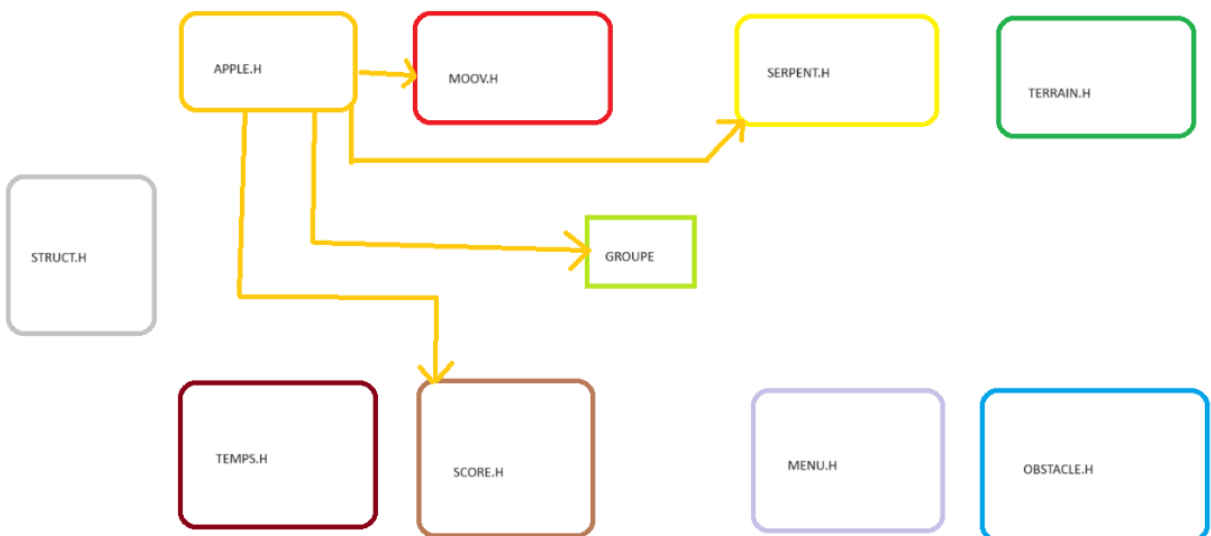


3. Le Makefile

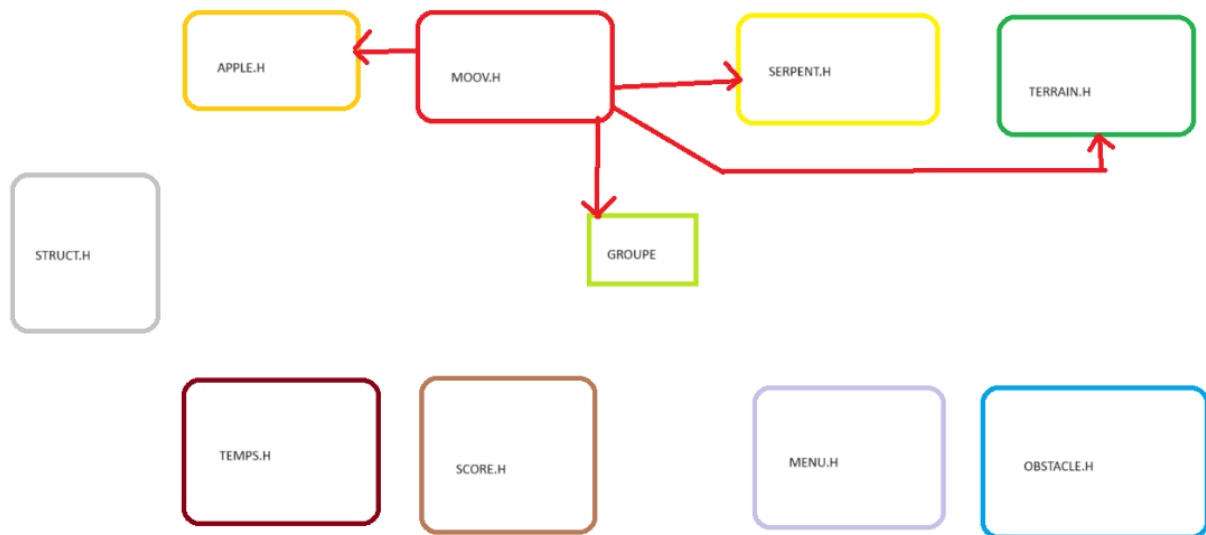
Pour ce projet comme dit précédemment, il a été fait avec un makefile ce qui permet de découper un programme en plusieurs petits programmes ce qui permet de mieux comprendre comment le programme agit ainsi que de pouvoir mieux trouver les parties à modifier si l'on en trouve le besoin. Voici une représentation graphique pour pouvoir visualiser notre makefile :



Comme vous pouvez voir dans ce diagramme struct.h interagit avec tous les fichiers, il contient toutes les structures nécessaires au fichier.

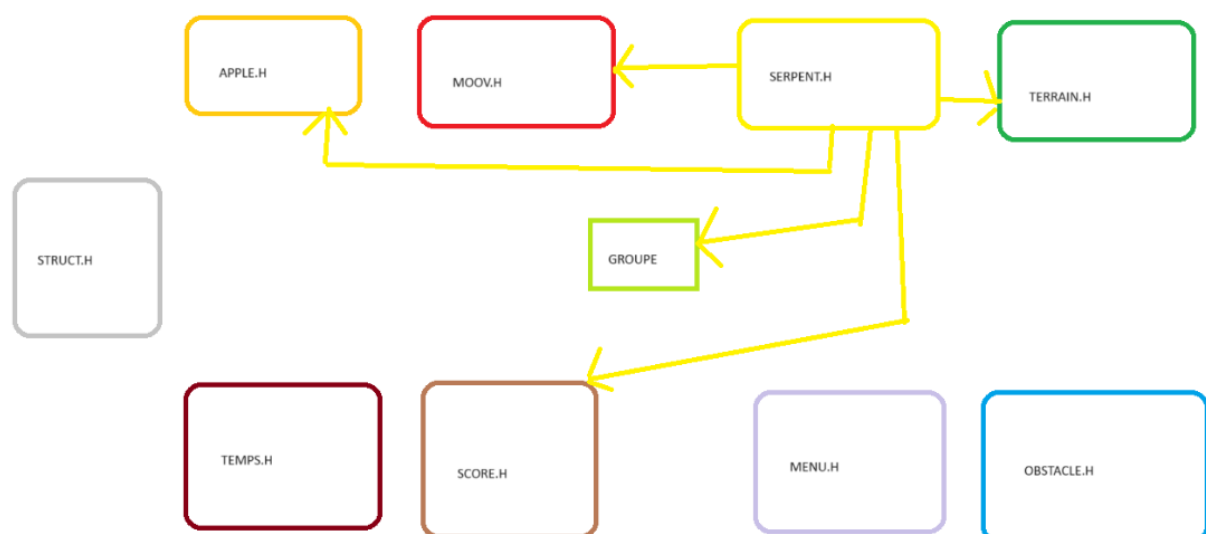


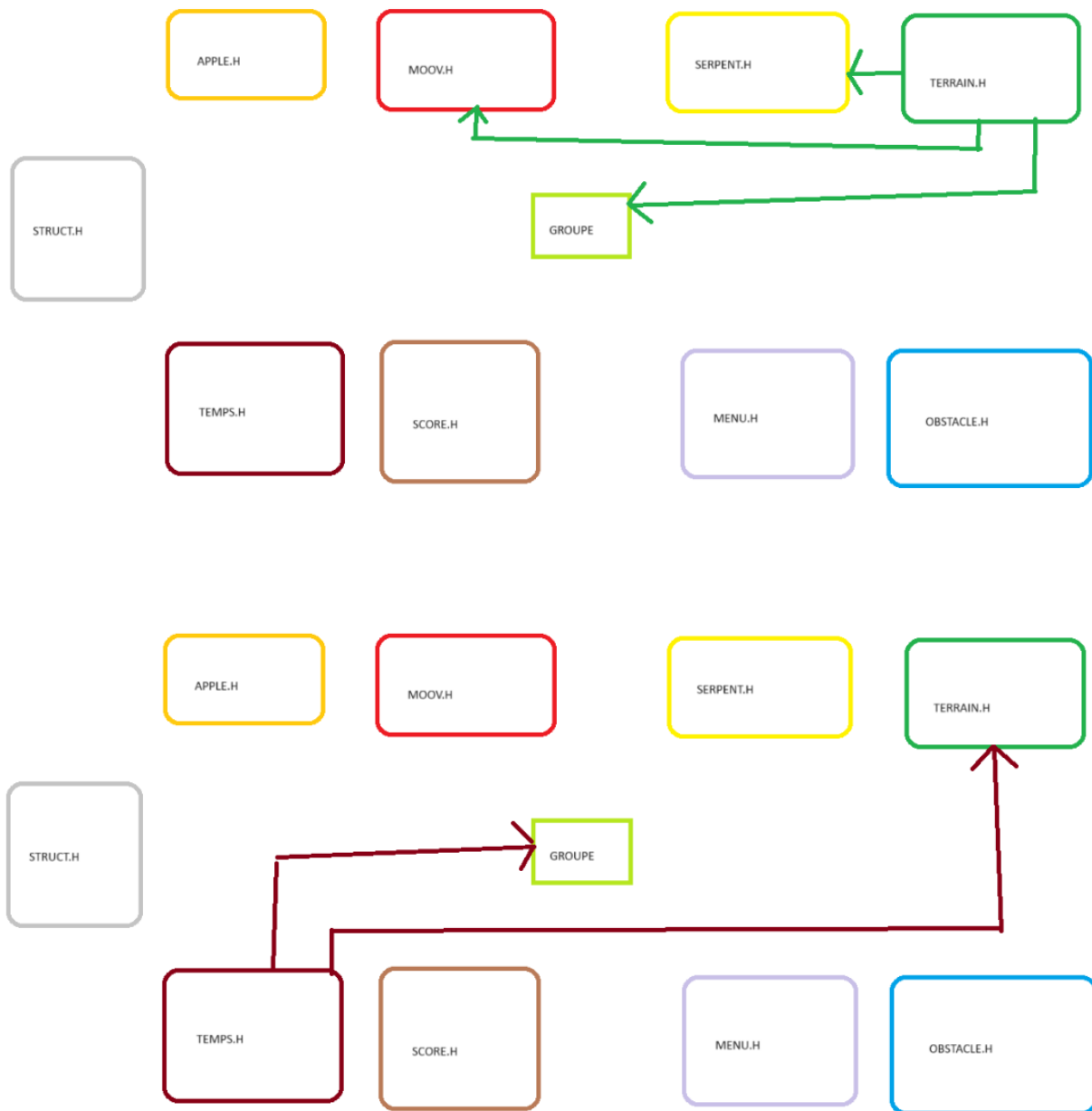
Apple.h interagit avec moov.h, serpent.h et score.h, car il est un indicateur pour le mouvement, il augmente la taille du serpent, il augmente le score s'il est en collision avec le serpent, et il fait partie de groupe, car groupe englobe tout le programme.

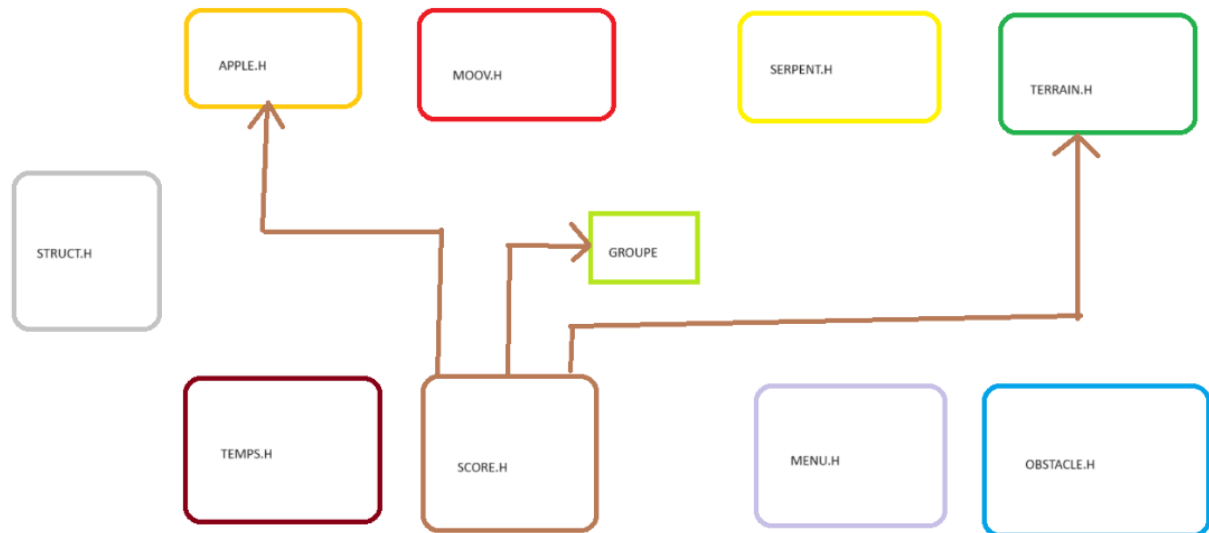


Moov.h interagit avec serpent.h, terrain.h, apple.h, et groupe.h car le moov.h change la direction du serpent, prend en compte les collisions avec le bord du terrain ainsi que des obstacles, il prend aussi en compte la collision avec les pastilles, et il est influent dans groupe.h.

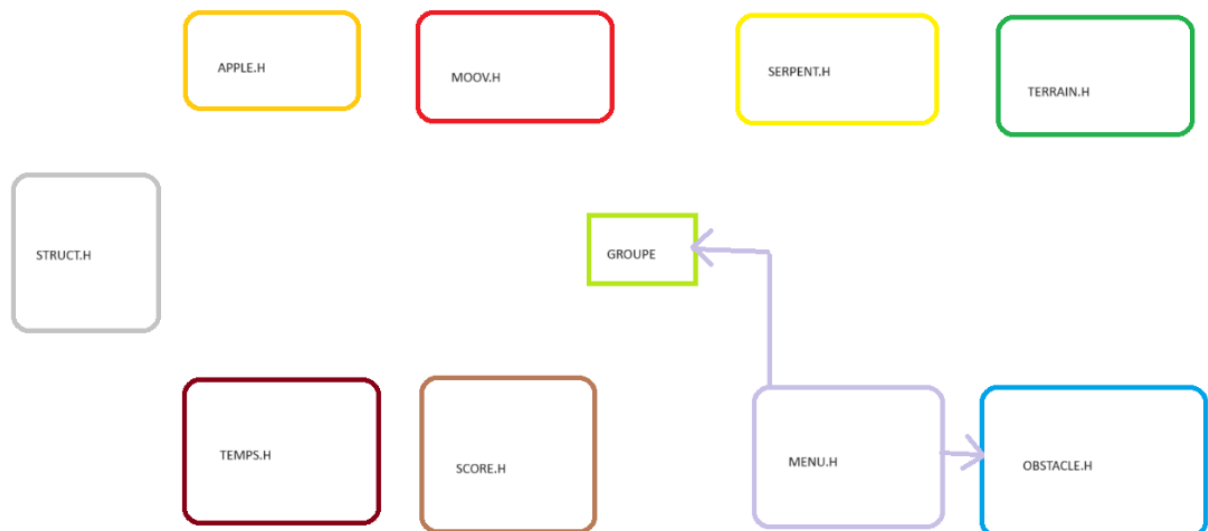
Par la suite, je ne vais pas détailler les échanges, car c'est toujours la même chose, c'est-à-dire des échanges de données entre les différents fichiers.h.

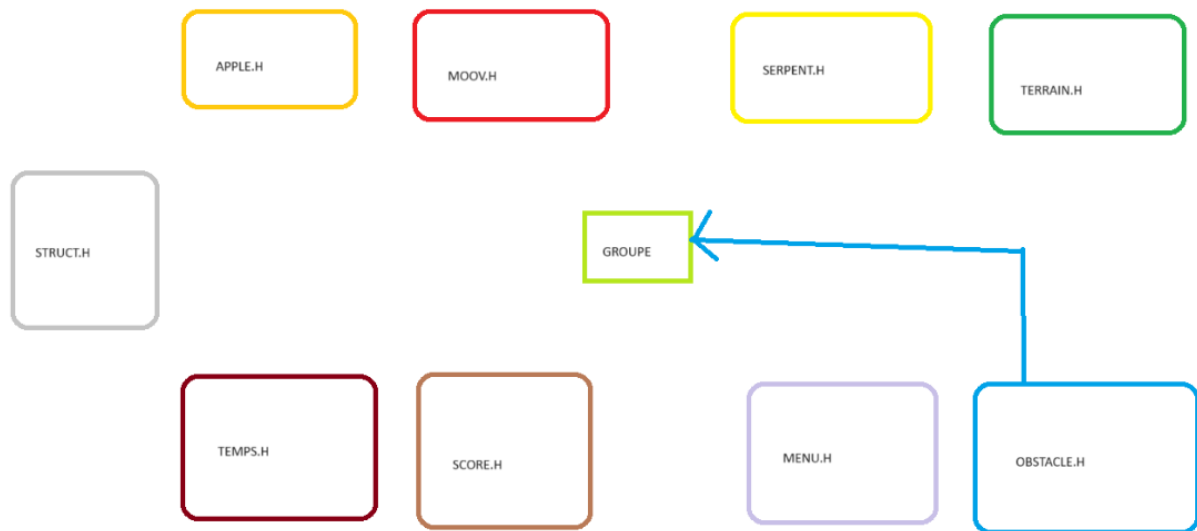






Dans notre makefile groupe n'a pas de .h, car c'est celui qui centralise les fichiers il ne fait que recevoir les données sans en rendre.





Pour conclure sur notre découpage, ce découpage est utile pour bien comprendre quelle partie du code de notre projet fait quoi, nous savons que dans obstacle.h il y a la gestion des obstacles, dans temps.h il y a la gestion du chrono, dans moov.h il y a tout le code en rapport avec le mouvement du serpent.

4. Les changements du serpent

Dans notre code, le serpent est défini en segment et ses segments prennent la place de celui qui est juste avant lui et qui a déjà été occupé par un segment et pour la tête du serpent elle est définie par des coordonnées qui augmentent de 1 par intervalle régulier par défaut les coordonnées qui augmentent sont les coordonnées X mais quand le joueur appui sur une des flèche directionnels la tête du serpent augmentera ses coordonnées en fonction de cette direction cependant cela fonctionne que pour les directions du haut et de droite comme les coordonnées d'ancrage (0,0) est à l'emplacement de départ du serpent donc quand le joueur appui sur la flèche directionnelle gauche ou bas ce n'est pas une augmentation de coordonnées, mais une diminution de coordonnées ce qui lui permet de se déplacer dans toutes ces directions tout en prenant en compte la collision avec son propre corps.

5. Conclusions des auteurs

Mathys TORREBLANCA :

Ce projet a été d'une aide dans ma compréhension du C89, et vu la finalité du projet je ne peux qu'être fier du travail fait avec mon camarade. Avec certains problèmes rencontrés durant le projet, ça a été vraiment intéressant d'avoir un autre point de vue et qui puissent aider à résoudre certains problèmes que je n'arrivais juste pas à résoudre seul. Au vu du travail donné, la répartition des tâches du projet a été bien faite ce qui m'a donné du temps pour travailler dans d'autres SAE comme celui de la machine virtuelle en SCR.

Ozvann ABRAHAM :

Ce projet est très intéressant, car j'avais déjà tenté de faire un jeu similaire, mais sous un autre langage de programmation sans le finir, car plusieurs problèmes se présentaient, mais en C89 ce projet était tout de suite plus réalisable surtout avec l'aide d'un camarade. J'ai pu remarquer que d'avoir ma propre machine virtuelle d'installer sur mon ordinateur était bien utile, mais présentait quelques problèmes comme par exemple : toute l'installation des différentes bibliothèques, le compilateur, ainsi que le makefile qui ne voulais pas fonctionner sur ma machine virtuelle, qui en revanche fonctionnait correctement une fois repasser sur la machine de l'IUT. En ayant ma propre machine virtuelle travailler à distance était plus abordable, car je n'avais pas besoin de connexion internet comme avec les machines virtuelles de l'IUT qui eux en ont besoin.