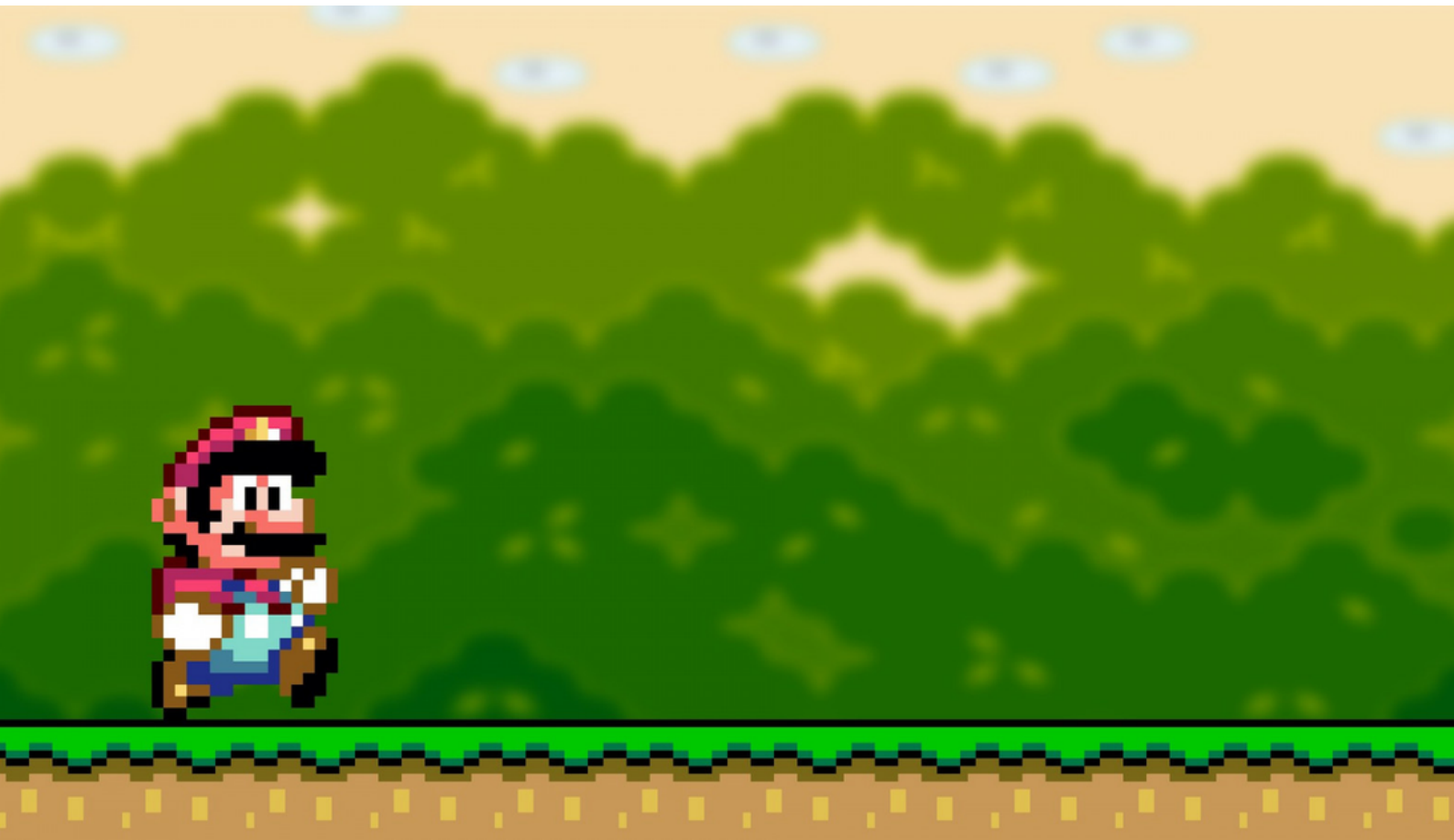


SAE11\_2022

# **JEU DE PAIRES EN C**



# SOMMAIRE

- 01** Introduction
- 02** Fonctionnalités
- 03** Choix du découpage et Makefile
- 04** Gestion de la grille de jeu
- 05** Conclusions personnelles

# INTRODUCTION

Le but du jeu de paires est de retrouver, à l'aide de sa mémoire, le bon emplacement de paires de cartes disposées aléatoirement dans une grille de jeu.

Le jeu fonctionne de la manière suivante :

- Au début de la partie, toutes les cartes sont cachées. Une carte se retourne quand le joueur clique dessus.
- Si le joueur clique sur deux cartes qui ne sont pas identiques, alors elles se retournent après un laps de temps d'une seconde.
- Si le joueur clique sur deux cartes identiques, elles restent retournées durant le restant de la partie.
- Le jeu se termine lorsque le joueur a trouvé l'ensemble des paires de la grille.

Le but de cette SAé était de programmer ce jeu avec le langage C à l'aide de la bibliothèque graphique de l'IUT.

Nous avons choisi, pour l'aspect visuel de notre jeu, de nous inspirer d'un jeu de la saga Mario, nommé ici "Super Jeu de Paires".



Voici l'exemple de l'interface de notre jeu, lorsqu'on le lance pour la première fois à l'aide de notre Makefile (se référer à l'onglet "Choix du découpage et Makefile" dans le sommaire pour savoir comment le lancer).

# FONCTIONNALITES

Notre jeu de paires dispose des fonctionnalités suivantes :

- Lorsque la partie se lance, un chronomètre est affiché en haut à droite de la fenêtre :

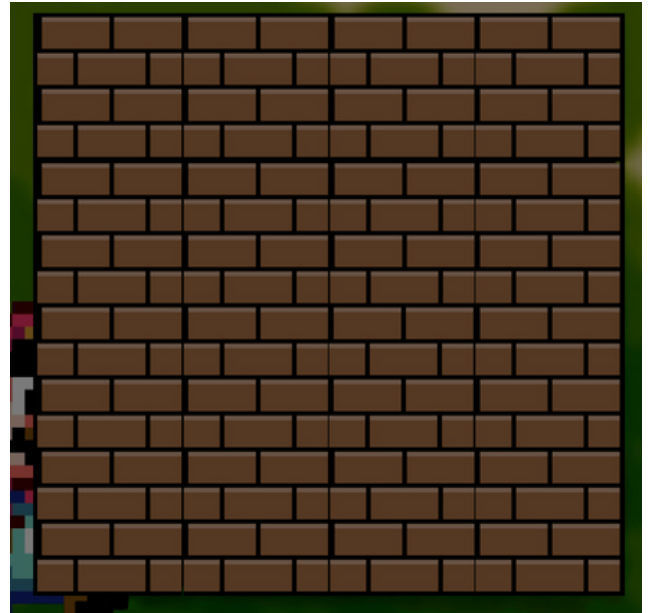
temps: 00:03

- Le joueur a le choix de jouer sur des grilles de tailles différentes.

Nous avons opté pour trois différents niveaux : le niveau 1 (facile, 4x4), le niveau 2 (moyen, 5x4) et le niveau 3 (difficile, 7x4).



- Un mode tricheur est accessible en appuyant sur la touche "t". Il permet d'afficher toutes les cartes de la grille de jeu. Si le joueur ré-appuie sur la touche "t", le mode tricheur se désactive et la grille retourne à l'état dans laquelle elle était avant l'activation du mode. Tant que le mode tricheur est activé, le chronomètre est désactivé.



# CHOIX DU DECOUPAGE & MAKEFILE

La création du Makefile a été le premier point abordé lors de la réalisation du projet. Le fichier Makefile permet de réaliser plusieurs actions :

- compiler le jeu (commande make),
- compiler et lancer le jeu (commande make run),
- supprimer les fichiers générés lors de la compilation (make clean).

Avant de commencer le projet, il a été important de réfléchir à la structure du code. Nous avons choisi de découper le programme en trois fichiers source : main.c, interface.c et grille.c.

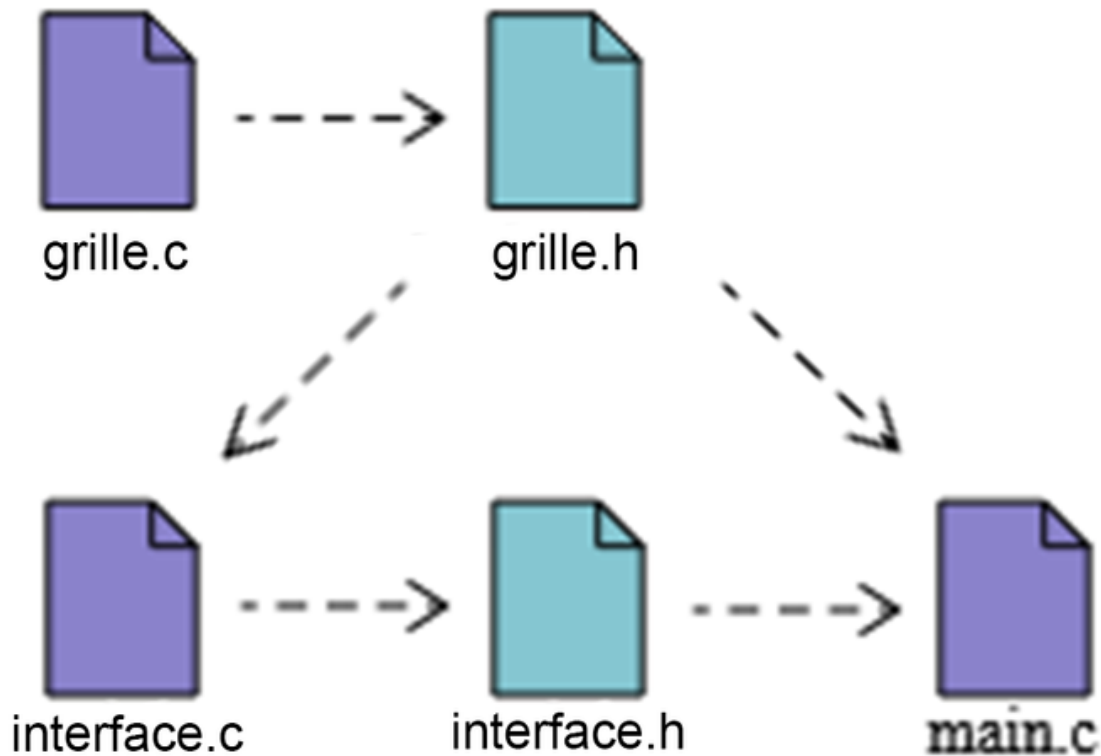
Le fichier source grille.c contient l'ensemble des fonctions permettant de créer une grille de jeu : génération et mélange de la grille, positionnement des cases et chargement des images.

Le fichier source interface.c contient des fonctions liées à l'affichage du jeu : l'affichage des différents menus et de la grille.

Pour finir, le fichier source main.c permet l'affichage de la fenêtre et appelle ensuite les fonctions des fichiers grille.c et interface.c. Pour que le fichier main.c puisse utiliser les fonctions définies dans ces autres fichiers, il est important d'inclure les fichiers header (.h) correspondants.

Ce découpage permet de se retrouver plus facilement dans le code et mieux organiser le développement.

Nous avons aussi séparé les dossiers contenant les fichiers sources et les images, permettant une meilleure organisation pour le dépôt de notre travail sur le serveur Git.



Voici le schéma sous forme de diagramme présentant les différents fichiers sources et leur dépendances.



# GESTION GRILLE DE JEU

Notre grille de jeu est représentée par un tableau d'entiers . Ces entiers correspondent aux numéros des images à afficher dans les cases. Également, deux tableaux d'entiers permettent de stocker les coordonnées X et Y de chaque case, ce qui permet de les afficher dans la fenêtre.

La première étape pour générer une grille de jeu est de créer un tableau d'entiers dans lequel on double le numéro de chaque image, pour représenter les paires. Pour ce faire, on initialise un compteur. Ensuite, on réalise une boucle allant de zéro jusqu'à  $n$ ,  $n$  étant la taille de la grille choisie par le joueur. Cette boucle avance de deux en deux et remplit la case de la grille à l'index actuel et la suivante avec la valeur du compteur, puis incrémente ce dernier. Cela permet ainsi de générer un tableau avec des numéros de cases doublés.

La seconde étape est de mélanger aléatoirement la grille générée précédemment. Pour ce faire, la fonction de mélange réalise une boucle allant de zéro jusqu'à  $n-1$ ,  $n$  étant la taille de la grille choisie par le joueur. A chaque fois que la boucle se répète, un entier compris entre 0 et l'index actuel est généré aléatoirement. On échange ensuite les valeurs stockées dans la grille entre l'index actuel et l'entier qui vient d'être généré.

# CONCLUSIONS PERSONNELLES

Baptiste: pour moi, ce projet étant le tout premier de ce semestre m'a beaucoup plu. Il m'a permis de voir mais surtout d'apprendre de nouvelles manières pour programmer, ainsi que de découvrir Visual Studio Code en profondeur. J'ai eu le plus de facilités avec les fonctions de la bibliothèque graphique de l'IUT, mais j'ai du passer beaucoup plus de temps avec les autres fonctions permettant la gestion du jeu et son interface. Me considérant comme ayant un niveau en C novice, j'ai préféré me lancer au début sur la programmation de l'interface, précisément l'affichage des boutons du jeu, des images et également de générer et mélanger la grille de jeu pour associer les sprites correspondants aux cases de celle-ci. La visualisation du jeu a été assez rapide, mais le plus compliqué pour moi était de savoir par où commencer pour associer mes différents choix avec les différentes parties du code que j'avais créé. Ce projet a été selon moi assez difficile car j'ai du créer des fonctions et les mettre en relation pour les faire fonctionner entre elles, et en utiliser beaucoup avec lesquelles j'ai eu du mal lors de mon apprentissage.

Samet: Ce premier projet sur la conception d'un jeu m'a beaucoup plu car, ayant comme l'un de mes projets professionnel de devenir développeur de jeux vidéos, cela m'a permis de comprendre comment initialiser un projet, réfléchir à la conception et à la structure du code. Ce projet m'a permis beaucoup de choses, j'ai pu installer une machine virtuelle sous Ubuntu sur mon ordinateur personnel pour pouvoir continuer à travailler sur le projet tranquillement. Pour ce projet, il fallait utiliser la bibliothèque graphique de l'IUT, qui est très simple d'utilisation, ce qui a rendu ce projet simple à faire, la seule difficulté a été de bien structurer le code et de réaliser des fonctions différentes pour organiser le code.