



Rapport du LOT_A_IPI2

Réalisé par

Ines Belkacem

Fatma Bali

Matheus Souza

Laabidi mayssa

Organisme : ENSIIE

ANNEE UNIVERSITAIRE

2016/2017

Introduction

Le but de ce projet est de réaliser un jeu appelé color_flood où nous allons mettre un nombre d'essais à ne pas dépasser, si vous le dépassez, vous serez éliminées de La partie qui se déroule en modifiant la couleur de la case située dans le coin en haut à gauche de la grille (origine) et de toutes celles qui appartiennent à la même composante 4-connexe que cette case.

Répartition des tâches :

Pour mieux comprendre le but de ce projet et pour se mettre d'accord sur les outils de développement à utiliser, nous avons fait plusieurs réunions. Nous avons utilisé l'outil TRELLO pour mieux gérer notre projet, où chacun de nous choisit les tâches dont il est responsable et indique son niveau d'avancement pour que tous les membres du groupe sache le niveau d'avancement.

solution :

Lot A :

Pour commencer à concevoir ce jeu, nous avons mis les bases de nos démarches constituent le premier lot A. Nous avons réservé tout d'abord dynamiquement un espace mémoire pour notre matrice carré réalisé par la fonction **faire_allocation_matrice ()** de **largeur** cases et chaque case est de taille maximum la taille d'un caractère et qui utilise la fonction prédéfinie **malloc ()**.

Ensuite nous avons essayé de remplir notre matrice de deux manières différentes, la première consiste de la remplir avec des valeurs générés aléatoirement c'est le résultat de la fonction **faire_saisie_matrice ()** où nous avons appelé une fonction prédéfinie **rand ()** qui permet de générer des valeurs aléatoires, et la deuxième consiste à la remplir à partir d'un fichier que nous l'avons rempli au préalable. Cette tâche est effectuée à l'aide de la fonction **lecture ()** qui prend en paramètre la grille et sa largeur et retourne cette grille remplie par le fichier texte initialisé de caractères.

Pour que nous puissions voir la matrice et afficher ses caractères nous avons développé une fonction **affiche ()** qui prendra en paramètre la grille et sa largeur et retourne les éléments de la grille.

Après, nous avons mis en place une fonction appelé **remplacer_matrice()**, cette fonction a pour but de réaliser le remplacement d'une case entrée en argument (paramètre x, paramètre y) par une nouvelle couleur entrée en paramètre.

Nous avons par la suite développé une fonction qui nous a beaucoup servi dans la suite du projet. Cette fonction est appelé **connexite_matrice()** qui prend en argument une grille de caractère ainsi que les coordonnées d'une case et le nombre des composantes connexes que nous l'avons initialisé à 0 et nous l'avons passé par adresses pour pouvoir extraire son dernier changement et sa dernière valeur. Cette fonction retourne le nombre des composantes connexes appelé nb et stocke leurs coordonnées dans un tableau a deux dimensions qu'on a alloué son espace dynamiquement.

Nous avons introduit finalement une dernière fonction **changement_couleur ()** qui prend en argument la matrice des couleurs, un caractère qui désigne la nouvelle couleur et des coordonnées quelconque d'une case d'origine. Si la couleur de la case d'origine et ses composantes connexes (dont nous avons extrait leurs coordonnées et les stocker dans le tableau à deux dimensions tab à l'aide de la fonction **connexite_matrice()**) est différente de la couleur de la nouvelle case entrée en paramètre et leurs couleurs changent et prennent la nouvelle couleur entrée en argument en utilisant la fonction **remplacer_matrice ()** mentionnée en haut du document.

Pour terminer cette partie nous avons libéré l'espace mémoire alloué pour éviter les pertes d'espace inutile.

Test

Pour réaliser nos testes et pour faciliter les taches d'exécutions, nous avons utilisé le Framework **CUNIT** permettant l'écriture et l'exécution de tests unitaires.

Documentation

Pour commenter et documenter nos codes et afin de structurer nos démarches, nous avons utilisé l'outil de documentation **DOXYHEN**

Conclusion

Comprendre comment développer le lot A, le fait de choisir les structures, fixer les types approprié nous a appris comment procéder à penser efficacement et utilement à la suite de notre projet.