

Rapport Labyrinthe

But du projet :

Le but du projet était de créer un labyrinthe en partant d'un quadrillage plein et en enlevant aléatoirement des murs du quadrillage.

Par la suite on rajoute des options tel que la seed pour l'aléatoire ou la taille du labyrinthe.

Développement du projet :

Pour commencer il a fallut choisir une structure pour représenter le labyrinthe, on a choisi pour cela la structure suivante :

```
typedef struct {  
    int x;  
    int y;  
} Case;
```

```
typedef struct {  
    int mur_hori;  
    int mur_vert;  
} Labyrinthe;
```

Ce choix a été fait car c'est la structure avec laquelle on se représentait le mieux les choses.

Il a ensuite fallut coder les fonctions de base permettant la représentation complète du labyrinthe dans le terminal puis avec l'aide des fonctions faites lors du TP sur Union-Find tout en l'adaptant au projet, on a pu implémenter la suppression aléatoire des murs de façon à obtenir le labyrinthe.

Cette partie terminée, il a ensuite fallut faire la partie MLV puis le rajout des différentes fonctions et de leurs gestions d'erreurs en fonctions des arguments rentrés par l'utilisateur.

Les fonctions *attente_valide()*, *taille_valide()* et *graine_valide()* sont les fonctions contrôlant respectivement les arguments *--attente=D*, *--taille=AxB*, *--graine=C* lors du lancement du programme, elles contrôlent si les arguments rentrés sont bon et si ceux-ci le sont, elles lance le programme avec les dits arguments sans problème. De même avec la fonction *Affichage_console()* qui est lancée sous certaines conditions.

Le projet a été réparti en deux partie :

- l'un c'est occupé de la base consistant a créer et représenter le labyrinthe dans le terminale.
- l'autre a fait la partie MLV ainsi que les options rajoutées au programme.

VONG Stéphane
BARDIN Nicolas

Compiler et exécuter le programme

La compilation du programme s'effectue avec :

gcc -Wall Lab_MLV.c -o Laby -IMLV

Le lancement se fait avec :

./Laby --mode=texte --taille=AxB --graine=C --attente=D

avec A et B des entiers positifs

C et D des entiers positifs ou nuls

Les options (qui sont les arguments commençant par --) sont optionnelles. On peut lancer le programme avec 0 argument tout comme on peut le lancer avec 4 arguments.