UVSQ université PARIS-SACLAY | ISTY
Institut des Sciences et Techniques des Yvelines
CAMPUS DE MANTES EN YVELINES
CAMPUS DE SAINT-QUENTIN-EN-YVELINES

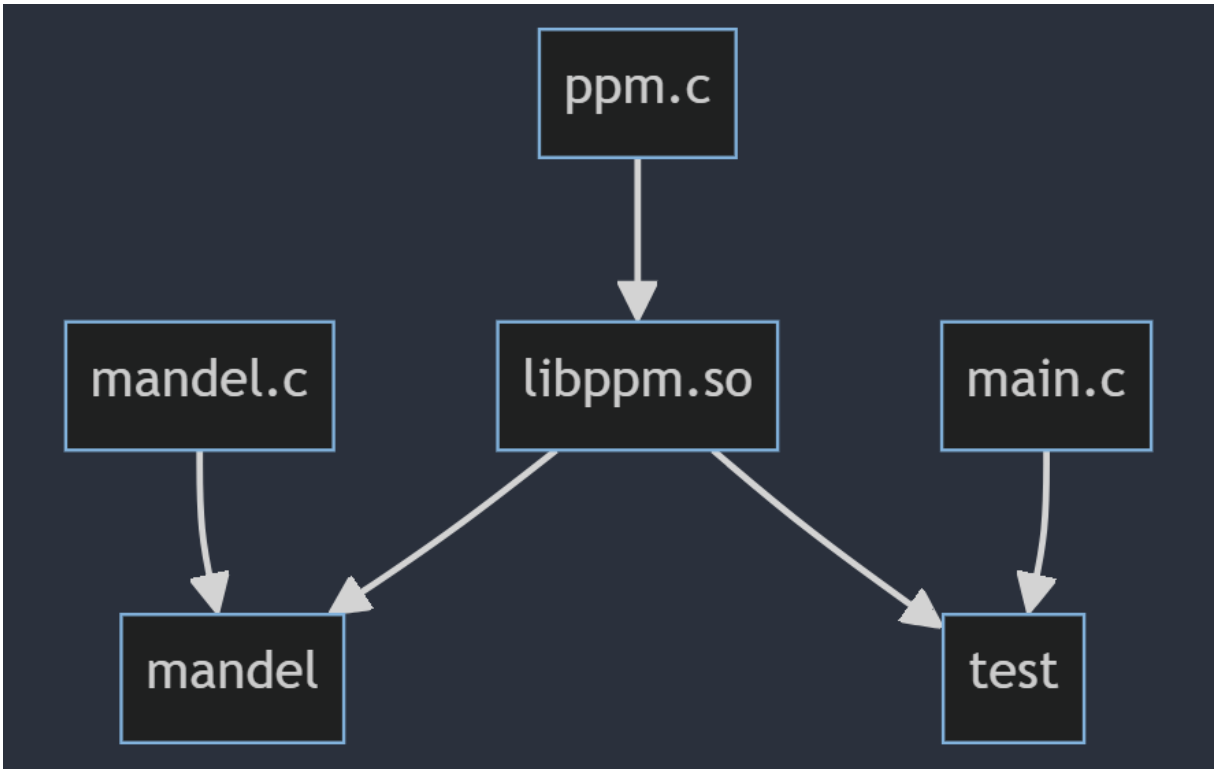
# RAPPORT TP GIT

**REALISE PAR HIND KHAYATI ET MOUHMED JIDDOU**

# Exercice 1:

## Question2

*La référence des fichiers :




## Question3 :

*Le fichier "mandel.c" contient le code principal pour créer l'ensemble de Mandelbrot. Il utilise des images au format PPM pour représenter visuellement l'ensemble de Mandelbrot. Les dimensions de l'image ainsi que les paramètres de l'ensemble de Mandelbrot sont définis comme des constantes. Les fonctions "cx" et "cy" sont employées pour calculer les parties réelle et imaginaire d'un nombre

complexe. Dans la fonction principale "main", l'ensemble de Mandelbrot est généré en parcourant chaque pixel de l'image et en appliquant l'algorithme spécifique à l'ensemble de Mandelbrot.

**ppm.c :**

Ce fichier implémente les fonctions nécessaires pour manipuler les images PPM.

Il inclut les fonctions pour initialiser une image PPM, libérer sa mémoire et l'écrire dans un fichier.

**ppm.h :**

Ce fichier contient les déclarations des structures et des fonctions associées aux images PPM.

Il définit la structure ppm_pixel pour représenter un pixel PPM et la structure ppm_image pour représenter une image PPM.

Les fonctions ppm_image_init, ppm_image_release, ppm_image_setpixel et ppm_image_dump sont déclarées pour manipuler les images PPM.

**Makefile :**

Ce fichier contient les règles pour compiler le code source et générer les exécutables.

Il définit les règles pour compiler les fichiers main.c et mandel.c avec les dépendances appropriées.

**main.c** (Non présent, mais peut être supposé) :

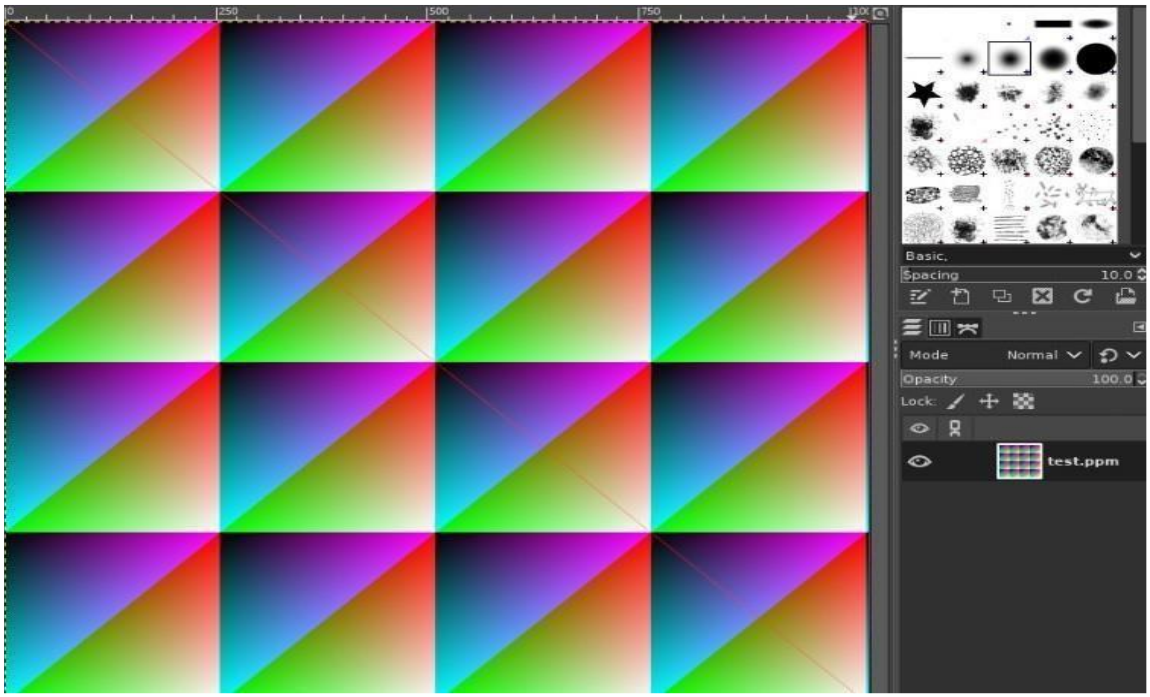
Ce fichier contient un programme de test pour créer une image PPM avec des pixels colorés de manière arbitraire.

Il est utilisé pour tester les fonctionnalités de base des images PPM.

Ces fichiers ensemble forment un ensemble de code pour générer et manipuler des images PPM, en particulier pour représenter l'ensemble de Mandelbrot.

## 4) La documentation du ppm.h et dans le fichier Faisons un test :

1) Make
2)./test
3) Gimp test



# Exercice 2:

**Question2**

1) La correction est faite dans les deux fichies mandel.c et Makefile

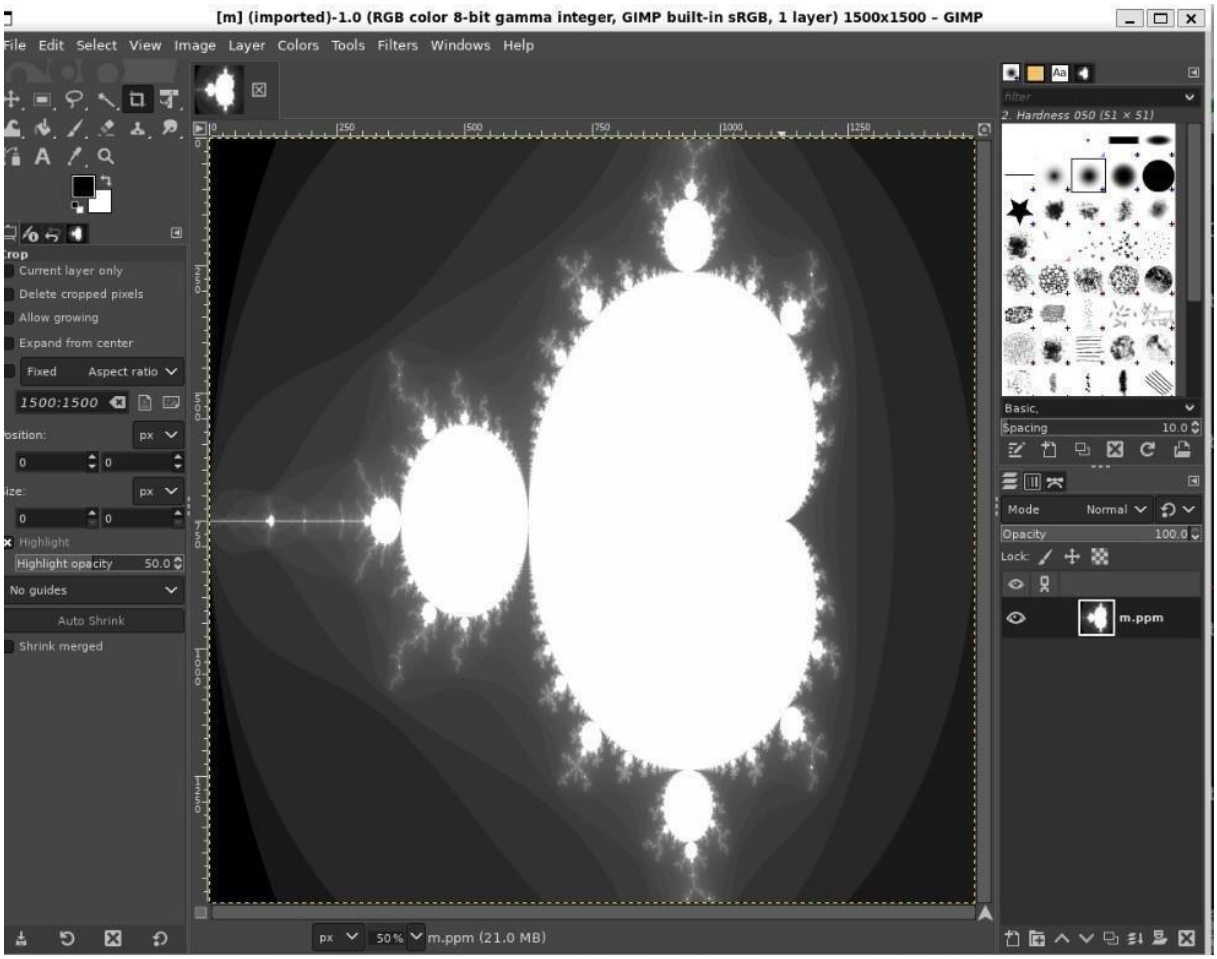
Les Commandes :

* Make

* ./mandel

* gimp m.ppm

2)



# 2) Explication du code mandel.c

a) Les lignes 1 à 5 incluent les bibliothèques nécessaires telles que stdio.h, complex.h, math.h, et ppm.h pour les fonctionnalités standard et la manipulation des images au format PPM.

b) Les lignes 7 à 10 définissent les constantes utilisées dans le programme, telles que TRSH (seuil pour tester la divergence), ITER (nombre maximum d'itérations), SIZEX, et SIZEY (dimensions de l'image).

c) La structure col, déclarée aux lignes 12 à 17, représente une couleur par ses composantes rouge, vert et bleu.

d) La fonction getcol, définie de la ligne 19 à 34, prend une valeur val et une valeur maximale max en entrée, et retourne une couleur basée sur ces valeurs. Dans cet exemple, elle produit un dégradé de bleu à blanc en fonction de la valeur val par rapport à max.

e) Les fonctions cx et cy (ligne 36-46) calculent les composantes réelles et imaginaires d'un nombre complexe en fonction des coordonnées x et y, respectivement, compte tenu du zoom et du centre spécifiés.

f) La fonction mandelbrot, définie de la ligne 48 à 73, génère l'ensemble de Mandelbrot en calculant les valeurs pour chaque pixel de l'image en fonction du centre, du zoom et des dimensions spécifiés.

g) La fonction main, définie à partir de la ligne 75, initialise l'image, appelle la fonction mandelbrot pour générer l'ensemble de Mandelbrot, puis enregistre l'image au format .

# Exercice 3:

Comment ramener cette branche au-dessus de la master

**A. Assurant nous que nous sommes sur la branche master**

* git checkout master

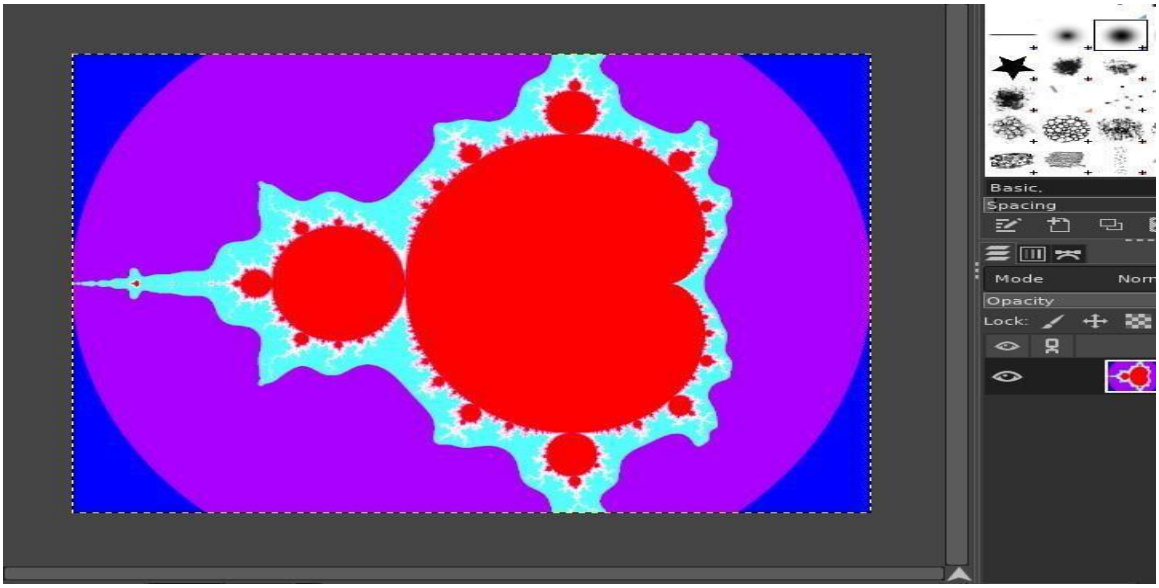
**B. Assurant Que la branche est à jour**

*git pull origin master

**C. on fait merge de branche color_support dans master**

**D.On résoud les conflits**

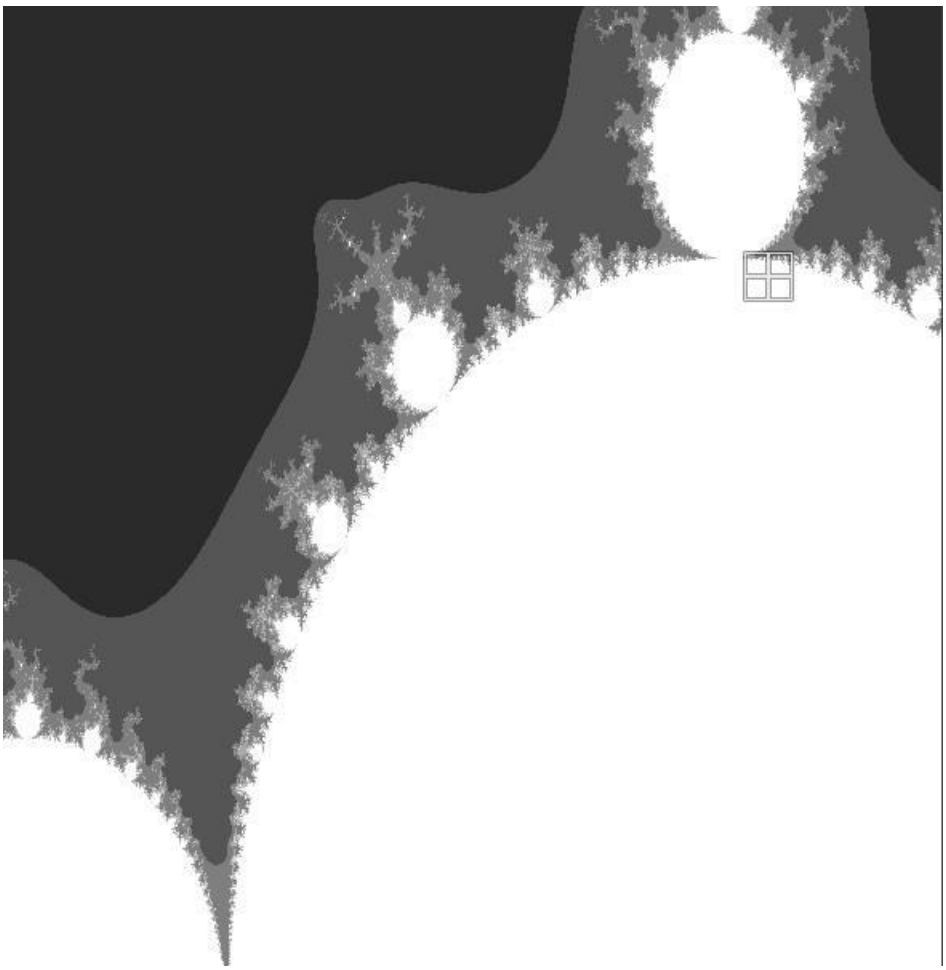

# LE RESULTAT



# Exercice 5 :

Idées pour améliorer le code :

•Implémentation du zoom



**Ajout de couleurs dynamiques : Créez des effets de dégradé de couleur ou d'animation pour rendre la fractale plus esthétique et expressive.**