

Projet d'Architecture Matérielle et Assembleur

ESGI – 2i
Soutenances 7 Janvier 2022

Ce projet devra être réalisé par groupe de trois.

Vous allez écrire un programme en assembleur x86-64 qui affiche à l'écran une fractale de Mandelbrot.

Vous utiliserez pour source de documentation le cours à l'adresse suivante :

<https://pub.phyks.me/sdz/sdz/dessiner-la-fractale-de-mandelbrot.html>

L'algorithme à utiliser est celui-ci (section algorithme du cours en ligne) :

```
// on définit la zone que l'on dessine. Ici, la fractale toute entière
définir x1 = -2.1
définir x2 = 0.6
définir y1 = -1.2
définir y2 = 1.2
définir zoom = 100 // pour une distance de 1 sur le plan, on a 100 pixels sur l'image
définir iteration_max = 50

// on calcule la taille de l'image :
définir image_x = (x2 - x1) * zoom
définir image_y = (y2 - y1) * zoom

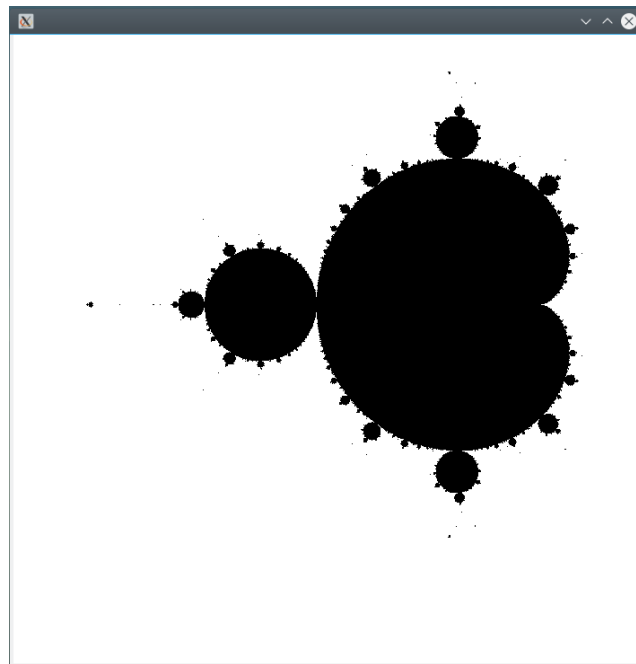
Pour x = 0 tant que x < image_x par pas de 1
  Pour y = 0 tant que y < image_y par pas de 1
    définir c_r = x / zoom + x1
    définir c_i = y / zoom + y1
    définir z_r = 0
    définir z_i = 0
    définir i = 0

    Faire
      définir tmp = z_r
      z_r = z_r*z_r - z_i*z_i + c_r
      z_i = 2*z_i*tmp + c_i
      i = i+1
    Tant que z_r*z_r + z_i*z_i < 4 et i < iteration_max

    si i = iteration_max
      dessiner le pixel de coordonnées (x; y)
    finSi
  finPour
finPour
```

Il génère une image de toute la fractale de Mandelbrot.

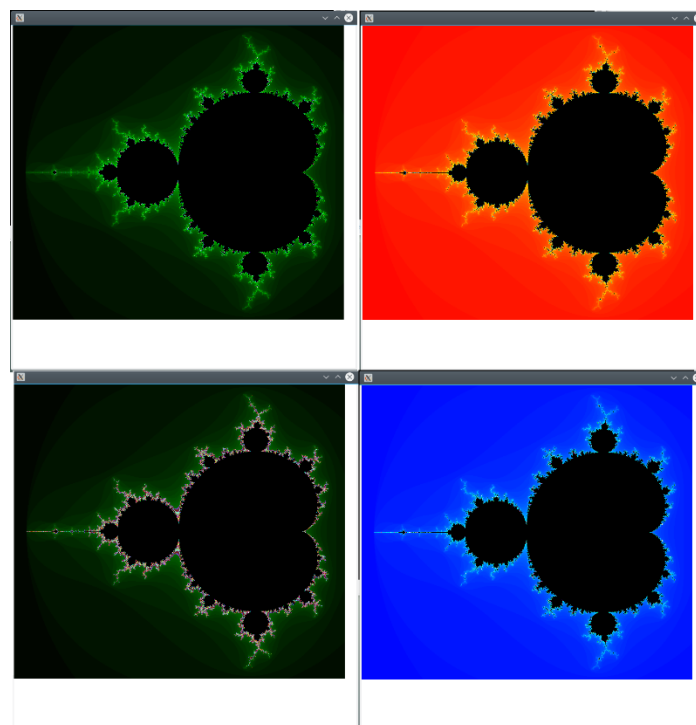
Son implémentation correcte est le minimum demandé pour ce projet :



Pour accéder aux meilleures notes, il vous faudra améliorer ce rendu avec deux choses :

1 - introduire des couleurs dans le dessin selon la nature des points (voir cours en ligne)

Quelques exemples :



2 - possibilité de zoomer sur une portion de la fractale (voir cours en ligne) en dur dans le code, puis en indiquant sur la ligne de commande la zone à dessiner.

Je vous fournis pour ce projet, sur MyGES, un code assembleur 'code_pour_dessiner.asm' qui ouvre une fenêtre et montre comment dessiner des lignes et comment changer leur couleur

Votre code final devra être écrit en assembleur NASM 64 sous Linux.

Lors de la soutenance, vous montrerez le résultat de votre travail ainsi que le code source commenté.