



Projet d'Infographie

fract'ol

42 staff staff@42.fr

Résumé: Ce projet consiste à créer graphiquement de jolies fractales.

Table des matières

I	Préambule	2
II	Sujet	3
III	Sujet - Partie bonus	5
IV	Consignes	6

Chapitre I

Préambule

Voici ce que Wikipédia a à dire sur la fracturation hydraulique :

La « fracturation hydraulique » est la dislocation ciblée de formations géologiques peu perméables par le moyen de l'injection sous très haute pression d'un fluide destiné à fissurer et micro-fissurer la roche. Cette fracturation peut être pratiquée à proximité de la surface, ou à grande profondeur (à plus de 1 km, voire à plus de 4 km dans le cas du gaz de schiste), et à partir de puits verticaux, inclinés ou horizontaux.

Cette technique relativement ancienne (1947), inventée pour les gisements d'hydrocarbures conventionnels, a vu son intérêt renouvelé par son association au forage horizontal (développé, lui, à partir de 1980). C'est la maîtrise graduelle de la rentabilité économique de cette association pour les gisements non-conventionnels, qui a guidé le développement récent de l'exploitation de ces derniers : elle a rendu accessibles des ressources autrefois soit inaccessibles, soit qui n'auraient été exploitables qu'à des coûts exorbitants et avec lenteur.

Elle est effectuée en fracturant la roche par un « stress » mécanique³ à l'aide d'un fluide injecté sous haute pression à partir d'un forage de surface, pour en augmenter la macro porosité et moindrement la micro-porosité. Le fluide peut être de l'eau, une boue ou un fluide technique dont la viscosité a été ajustée.

Ce projet ne s'appelle pas *fract'oil* et par conséquent n'a aucun rapport avec la fracturation hydraulique.

Chapitre II

Sujet

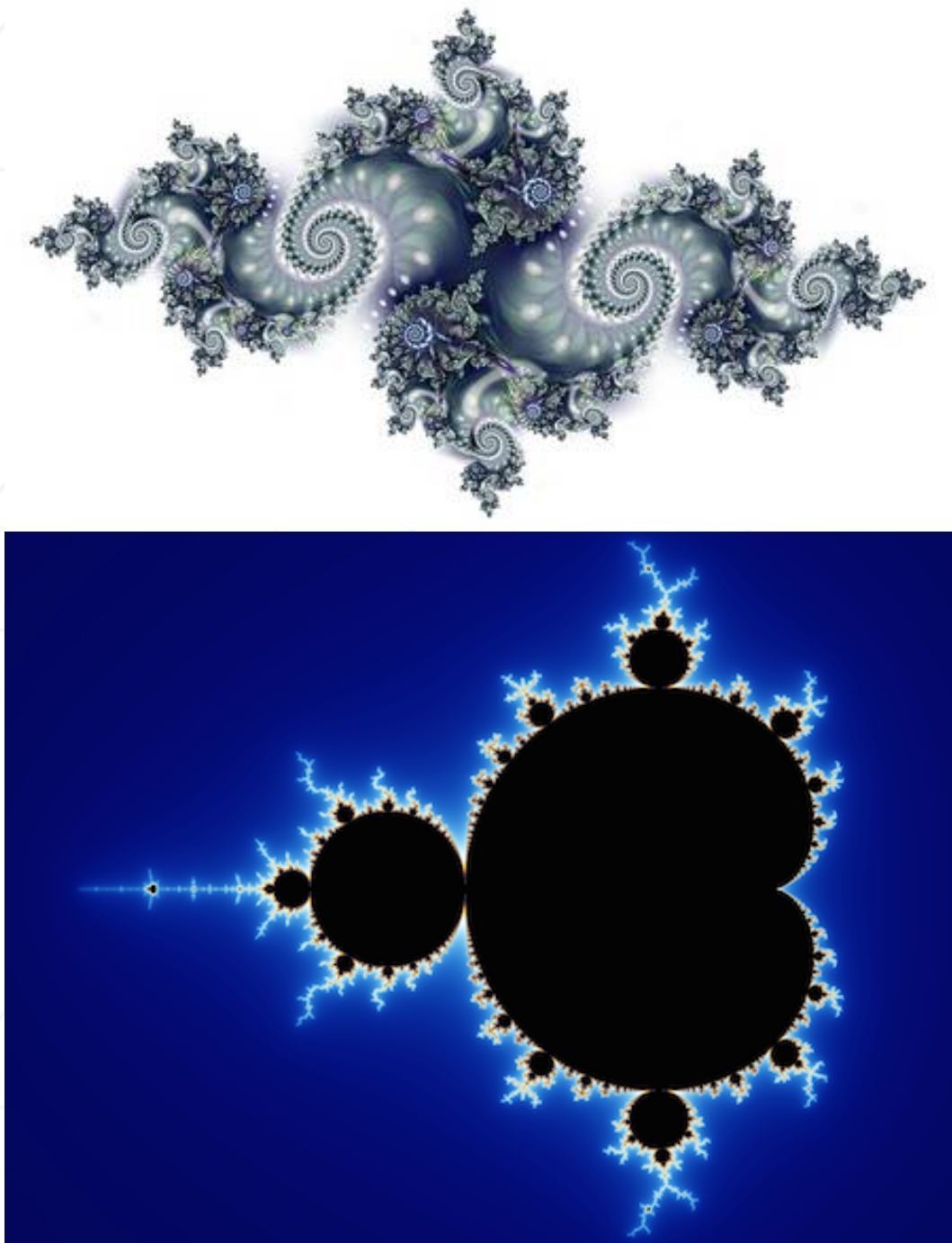
Ce projet consiste à créer un petit logiciel d'exploration fractale. Commencez donc par voir ce qu'est une fractale.

Les contraintes sont les suivantes :

- Votre logiciel doit proposer au minimum 3 types de fractales différentes, dont au moins l'ensemble de Julia et l'ensemble de Mandelbrot (soit au moins une troisième fractale au choix).
- Il doit être possible de faire varier avec la souris (sans clic) le paramètre de l'ensemble de Julia. Pour les autres types de fractale, cela est laissé à votre discrétion.
- La molette de la souris permet de zoomer et dézoomer sur la position actuelle de la souris, et cela de façon quasi infinie (modulo les limites de la machine). C'est dans le principe même des fractales.
- Il doit y avoir un jeu de couleur minimum pour ressentir la profondeur de chaque fractale. Mieux : lâchez-vous sur les effets psychédéliques.
- Un paramètre est passé en ligne de commande pour définir quel type de fractale est à afficher. Si il n'y a pas de paramètre fourni, ou si le paramètre est invalide, le programme affiche la liste des paramètres disponibles et quitte.

En ce qui concerne la représentation graphique :

- Vous devez utiliser la `minilibX`, dans sa version pour XQuartz installée par défaut sur les dump, ou bien dans sa version native pour MacOS X.
- Vous devrez gérer l'`expose` correctement.
- La touche ESC permettra de quitter le programme.
- L'utilisation des images de la `minilibX` est fortement conseillée.



Chapitre III

Sujet - Partie bonus

Voici quelques idées de bonus intéressants à réaliser. Vous pouvez évidemment ajouter des bonus de votre invention, qui seront évalués à la discrétion de vos correcteurs.

- En plus du zoom : déplacement avec les flèches.
- Faire tourner la palette de couleur.
- Plein de fractales sympa (des sites recensent plus d'une centaine de type différents).
- Représentations en 3D.
- Deux paramètres valides en ligne de commande, deux fenêtres pour deux fractales.

Chapitre IV

Consignes

- Ce projet ne sera corrigé que par des humains. Vous êtes donc libres d'organiser et nommer vos fichiers comme vous le désirez, en respectant néanmoins les contraintes listées ici.
- L'exécutable doit s'appeler **fractol**.
- Vous devez rendre un Makefile.
- Votre Makefile devra compiler le projet, et doit contenir les règles habituelles. Il ne doit recompiler le programme qu'en cas de nécessité.
- Si vous êtes malin et que vous utilisez votre bibliothèque **libft** pour votre **fractol**, vous devez en copier les sources et le **Makefile** associé dans un dossier nommé **libft** qui devra être à la racine de votre dépôt de rendu. Votre **Makefile** devra compiler la librairie, en appelant son **Makefile**, puis compiler votre projet.
- Pareil pour la MinilibX native MacOS.
- Vous ne devez pas avoir de variables globales.
- Votre projet doit être à la Norme.
- Vous devez gérer les erreurs de façon raisonnée. En aucun cas votre programme ne doit quitter de façon inattendue (Segmentation fault, etc...).
- Vous devez rendre, à la racine de votre dépôt de rendu, un fichier **auteur** contenant votre login suivi d'un '\n' :

```
$>cat -e auteur
xlogin$
$>
```

- Vous avez le droit d'utiliser toutes les fonctions que vous voulez de la minilibX, de votre libft, de la lib math, et de la libC si il ne s'agit pas d'une fonction qui devrait se trouver dans votre libft.
- Vous pouvez poser vos questions sur le forum, sur jabber, IRC, ...
- Bon courage à tous !