

# Des fractales en *Crocodile BASIC* (I)

## 1. Introduction

Cette série d'articles est consacrée au calcul de [figures fractales](#) telles que les célèbres ensembles de Mandelbrot et de Julia. Ces derniers sont créés par itération d'une fonction de nombre complexe,  $f(z)$ .

Dans cette première partie, nous nous limiterons à la fonction la plus classique :  $f(z) = z^2 + c$  où  $c$  est une constante complexe. Les programmes sont écrits dans le langage [Crocodile BASIC](#) qui est dérivé du [BASIC Amstrad](#) et du langage [PANORAMIC](#)

## 2. L'ensemble de Mandelbrot

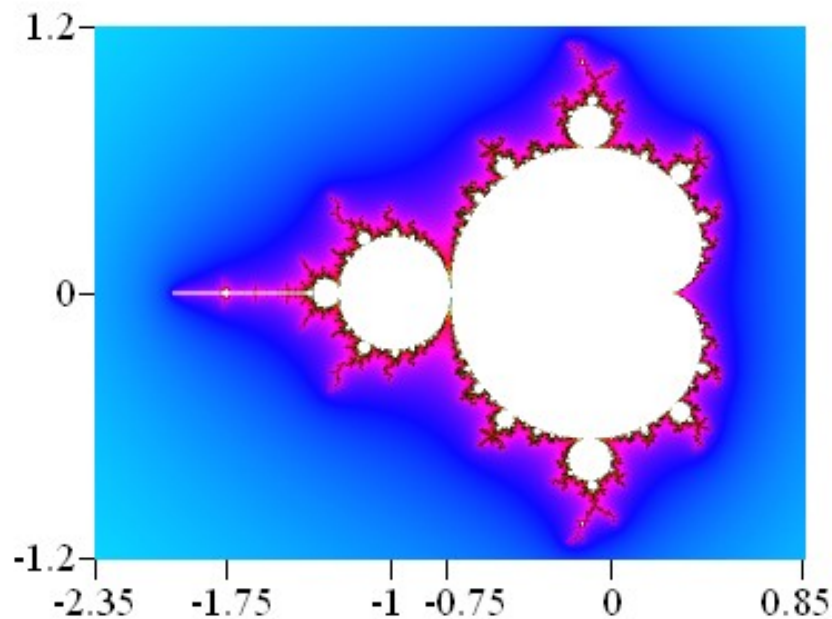
L'[ensemble de Mandelbrot](#), nommé d'après le mathématicien [Benoît Mandelbrot](#), est défini comme l'ensemble des nombres complexes  $c$  pour lesquels la suite :

$$z_0 = 0 ; \quad z_n = (z_{n-1})^2 + c$$

converge vers une valeur finie.

En pratique, la suite est évaluée en chaque point  $c$  du plan. Les points de l'ensemble sont colorés de la même couleur, et les points extérieurs à l'ensemble reçoivent une couleur fonction de leur vitesse de divergence.

Une vue générale de l'ensemble de Mandelbrot est donnée ici :



- L'ensemble de Mandelbrot est en blanc. Il comporte plusieurs parties :
  - une cardioïde constituant la partie principale de l'ensemble (courbe en forme de coeur, à droite). Le « creux » de la cardioïde se situe au point (0,25 ; 0)
  - un disque principal de rayon 0,25 centré en (-1 ; 0) et tangent à la cardioïde au point (-0,75 ; 0)
  - un prolongement rectiligne ou « antenne » (partie gauche de l'ensemble), situé sur l'axe Ox jusqu'à l'abscisse -2 et montrant un mini-ensemble de Mandelbrot à l'abscisse -1,75

L'ensemble est hérissé de « bourgeons » de taille variable émettant des filaments. Les plus gros de ces bourgeons se trouvent aux deux pôles de la cardioïde. Chaque bourgeon est lui-même recouvert d'autres bourgeons, tandis que les filaments donnent naissance à de petites copies plus ou moins déformées de l'ensemble. Ces structures se répètent à toutes les échelles.

Les mathématiciens ont montré que :

- l'ensemble de Mandelbrot est entièrement inclus dans le cercle de rayon 2 : les nombres complexes de module supérieur à 2 ne font donc pas partie de l'ensemble.
- l'ensemble est connecté, c'est à dire d'un seul tenant : toutes ses parties sont reliées par des filaments, ceux-ci pouvant être infiniment fins, donc invisibles sur les images.
- Le voisinage immédiat de l'ensemble est représenté en couleurs sombres, selon la méthode de l'*estimateur de distance* que nous étudierons plus loin.
- L'extérieur de l'ensemble est constitué par les points pour lesquels la suite tend vers l'infini. Il est divisé en bandes de couleurs, selon le nombre d'itérations que la suite demande pour atteindre une valeur donnée appelée *rayon d'échappement*. Plus on se rapproche de l'ensemble, plus le nombre d'itérations augmente, et plus il augmente rapidement, de sorte que les bandes deviennent de plus en plus étroites.

### **3. Les ensembles de Julia**

Ces ensembles, nommés d'après le mathématicien [Gaston Julia](#), sont définis d'une manière analogue à l'ensemble de Mandelbrot, sauf qu'ici le paramètre  $c$  est constant et la suite est initialisée avec les coordonnées  $z_{\text{pixel}}$  du point considéré :

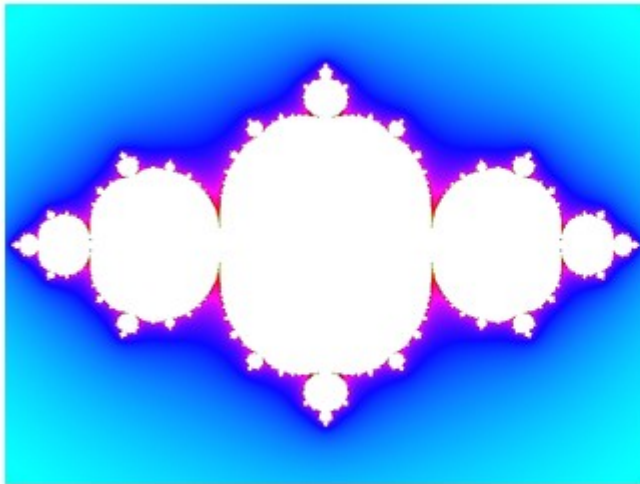
$$z_0 = z_{\text{pixel}} \quad ; \quad z_n = (z_{n-1})^2 + c$$

Il y a donc un ensemble de Julia pour chaque valeur du paramètre  $c$ .

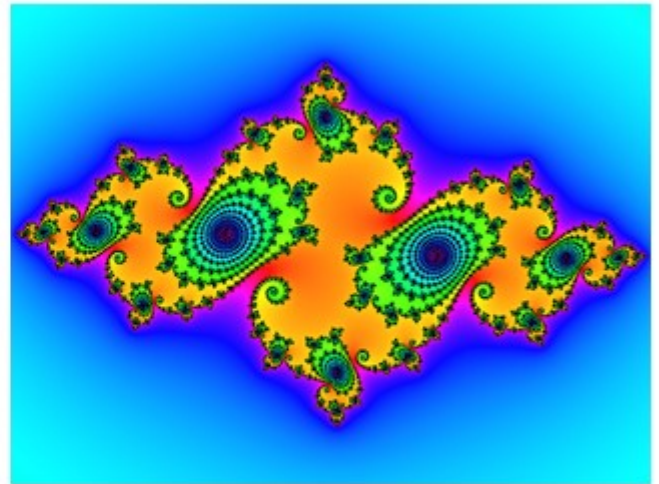
On montre que :

- Lorsque le point  $c$  appartient à l'ensemble de Mandelbrot, l'ensemble de Julia est connecté, c'est-à-dire d'un seul tenant.
- Dans le cas contraire, l'ensemble de Julia est formé d'une infinité de parties semblables.

Exemples :



julia\_2\_a



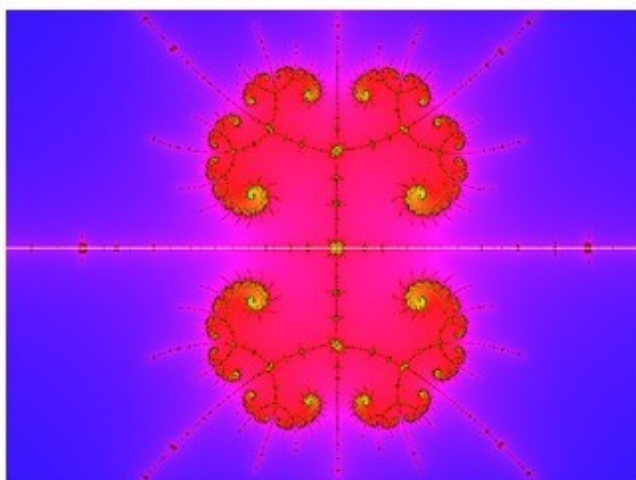
julia\_2\_b

- **julia\_2\_a** :  $c = (-0,75 ; 0)$  : ce point appartient à l'ensemble de Mandelbrot (il s'agit du point de jonction entre la cardioïde et le cercle principal) ; l'ensemble de Julia est connecté.
- **julia\_2\_b** :  $c = (-0,75 ; 0,1)$  : ce point est en-dehors de l'ensemble de Mandelbrot ; l'ensemble de Julia est discontinu.

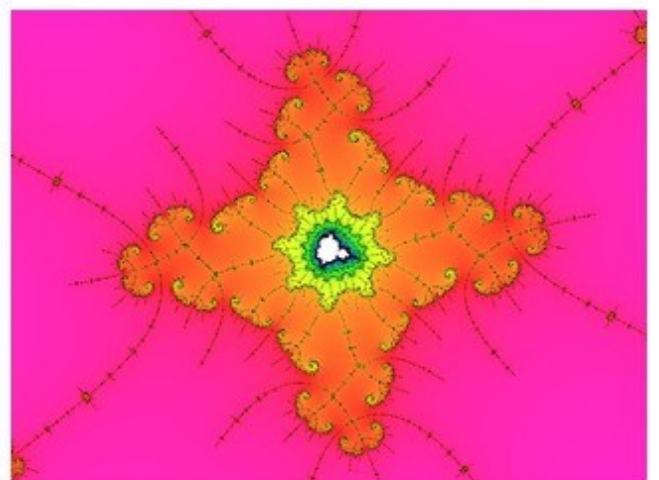
#### 4. Les EJS (« Embedded Julia Sets »)

Ce terme désigne des formations ayant l'apparence d'ensembles de Julia, mais situées à l'intérieur de l'ensemble de Mandelbrot, à proximité des mini-ensembles. Ils sont constitués de structures disposées symétriquement autour d'un « noyau » central, lequel contient lui-même un mini-ensemble de Mandelbrot.

Les images suivantes montrent un EJS situé sur l'« antenne » de l'ensemble de Mandelbrot. L'agrandissement montre un mini-ensemble à l'intérieur d'une des structures formant l'EJS.



antenna\_01



antenna\_02

## **5. Aspects théoriques**

### **5.1. Itération de la fonction complexe**

La suite  $z_n = (z_{n-1})^2 + c$  est évaluée jusqu'à ce que le module  $|z_n|$  soit supérieur au rayon d'échappement, ou que le nombre d'itérations soit supérieur à une valeur fixée à l'avance. Dans ce dernier cas, le point est considéré comme appartenant à l'ensemble.

### **5.2. L'estimateur de distance**

Comme son nom l'indique, ce paramètre estime la distance entre le point testé et l'ensemble. Pour le calculer, on utilise le fait que le nombre d'itérations requis pour atteindre le rayon d'échappement augmente de plus en plus vite lorsqu'on s'approche de l'ensemble. Cette vitesse est estimée par la dérivée de la fonction itérée.

- Pour l'ensemble de Mandelbrot, la fonction dépend du paramètre  $c$ . Il faut donc dériver par rapport à ce paramètre, ce qui donne :

$$z'_n = 2 z_{n-1} z'_{n-1} + 1 \quad \text{avec } z'_0 = 0$$

- Pour l'ensemble de Julia, le paramètre  $c$  est constant. La dérivée se réduit à :

$$z'_n = 2 z_{n-1} z'_{n-1} \quad \text{avec } z'_0 = 1$$

En fin d'itérations, l'estimateur de distance est donné par :

$$D = (2 |z_n| \ln |z_n|) / |z'_n|$$

Cette valeur est d'autant plus faible que l'on est proche de l'ensemble.

### **5.3. La méthode du *continuous dwell***

Comme nous l'avons vu précédemment, la coloration des points extérieurs à l'ensemble dépend du nombre d'itérations requis pour atteindre le rayon d'échappement. A proximité de l'ensemble, ce nombre varie beaucoup d'un point à l'autre et le mélange de couleurs qui en résulte donne à la figure un aspect brouillé. Pour y remédier, on peut « lisser » la distribution du nombre d'itérations en lui appliquant une transformation logarithmique. La formule utilisée dans nos programmes est la suivante :

$$\text{Dwell} = \text{Iter} + \log_2 ( \ln \text{Esc} / \ln |z_n| )$$

où  $\text{Iter}$  désigne le nombre d'itération et  $\text{Esc}$  le rayon d'échappement. La fonction  $\log_2$  est le logarithme de base 2, tel que  $\log_2 z = \ln z / \ln 2$

## 6. Aspects graphiques

### 6.1. Format des images

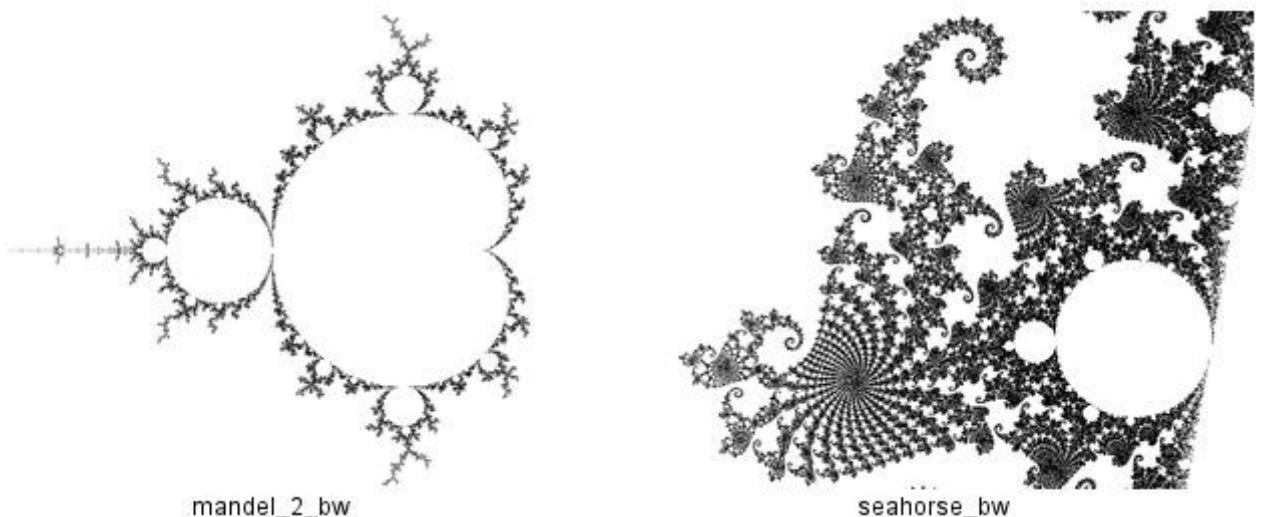
Les images générées par le programme ont par défaut une taille de  $800 \times 600$  pixels. La résolution est définie par un paramètre **ZoomFact**, tel que la valeur **ZoomFact** = 1 correspond à une échelle verticale de 4, soit une échelle horizontale de  $4 \times (800 / 600) \approx 5,33$ . L'échelle verticale pour une valeur quelconque de **ZoomFact** est donc  $4 / \text{ZoomFact}$ .

### 6.2. Coloration des images

L'aspect des images fractales dépend beaucoup de la méthode de coloration choisie. Nous utilisons une version simplifiée d'un [algorithme](#) décrit par Robert Munafo, et dont nous ne donnerons ici que les grandes lignes (les méthodes de coloration des fractales sont décrites en détail dans l'article IV de cette série).

- Les couleurs sont définies dans l'espace HSV (*Hue, Saturation, Value*)
- La luminosité (*Value*) est attribuée en fonction de l'estimateur de distance, de sorte que les points proches de l'ensemble apparaissent en couleurs sombres. Cela facilite la détection des parties les plus ténues de l'ensemble, qui peuvent ne pas apparaître en tant que telles. La luminosité est contrôlée par un paramètre **DistFact** et augmente lorsque ce paramètre augmente.
- La teinte (*Hue*) et la saturation sont attribuées en fonction du *continuous dwell*. La densité des couleurs est contrôlée par un paramètre **ColorFact** : plus ce paramètre est élevé (en valeur absolue) plus il y aura de couleurs sur l'image. Pour une image en noir et blanc, fixez **ColorFact** à zéro.

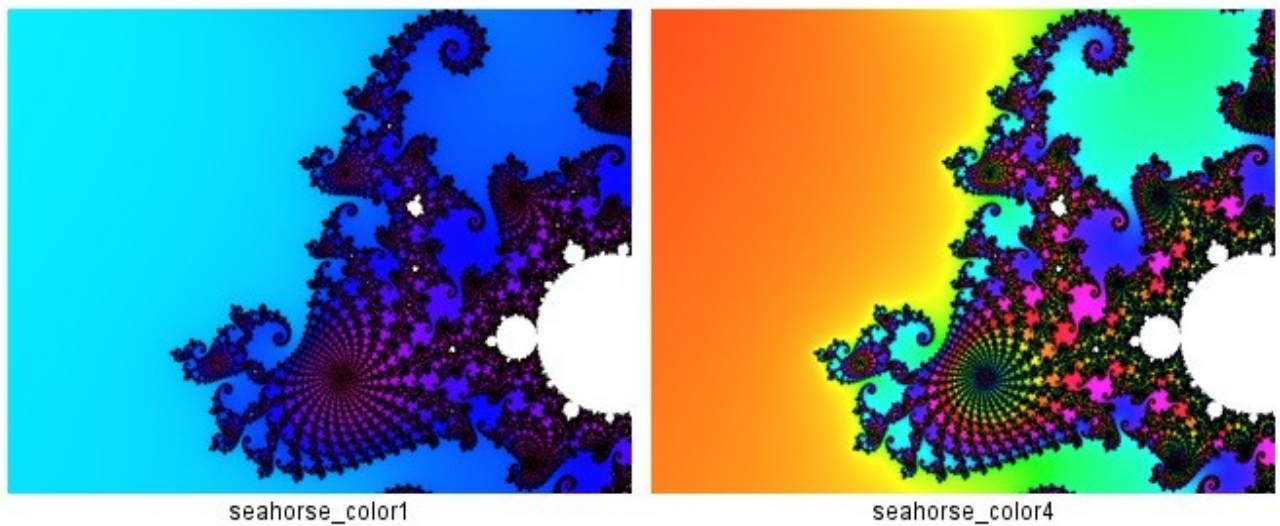
La figure suivante montre deux exemples d'images en noir et blanc : d'abord une vue générale de l'ensemble de Mandelbrot, suivie d'un agrandissement dans la « vallée des hippocampes » (ou *seahorse valley*), située entre la cardioïde et le disque principal.



Il peut être intéressant de commencer par créer l'image en noir et blanc, puis, lorsqu'on est satisfait du résultat, d'ajouter des couleurs en faisant varier progressivement le paramètre **ColorFact**.



A titre d'exemple, la figure suivante montre la même image de la *seahorse valley*, colorée avec **ColorFact** = -1 et **ColorFact** = -4 :



- Pour une meilleure apparence des bandes de couleurs, on peut dessiner une bande sur deux dans une teinte plus sombre. Cela se fait en attribuant une valeur positive à **ColorFact**.

Sur la figure suivante, la même image de la *seahorse valley* a été colorée avec **ColorFact** = 4 pour mettre en évidence les bandes de couleurs :



### 6.3. Symétrie par rapport à Ox

Il est démontré que, pour les fonctions de la forme  $z^p + c$  :

- l'ensemble de Mandelbrot est symétrique par rapport à l'axe Ox
- l'ensemble de Julia est symétrique si le nombre  $c$  est réel (voir l'exemple *julia\_2\_a*)

## 7. Les programmes : mandel.bas et julia.bas

Ces deux programmes sont inclus dans le logiciel *Crocodile BASIC* (répertoire `exemples\fractal`). Les codes sources étant fournis dans la distribution, nous ne commenterons ici que leurs principales caractéristiques.

### 7.1. Paramètres de base

Il s'agit des paramètres que l'utilisateur peut modifier :

- la taille de l'image et la couleur utilisée pour colorier l'intérieur de l'ensemble (blanc par défaut).
- le nom de l'image, les coordonnées (**x0**, **y0**) du centre de l'image, le nombre maximal d'itérations (**MaxIter**), les facteurs **ZoomFact**, **DistFact**, **ColorFact**, ainsi que les composantes **cx** et **cy** de la constante  $c$  pour l'ensemble de Julia ; ces paramètres sont passés à la procédure `init_params`.

Une série d'exemples est fournie sous forme d'appels à la procédure `init_params`

### 7.2. Programme principal

Pour tracer l'image on utilise la procédure `colormap` du *Crocodile BASIC*, à laquelle on passe l'adresse de la fonction `mandelbrot` qui calcule la couleur d'un point.

Le clic de souris permet de zoomer (bouton gauche = bouton 1) ou de dézoomer (bouton droit = bouton 2). La procédure `getcoord` recentre l'image sur la position de la souris. Les paramètres du tracé sont ajustés dans les procédures `SetZoom` et `SetParams`.

Une fois l'image complète, on peut la sauvegarder en pressant la touche S. L'image est sauvegardée dans un fichier `png`. Un fichier de type CSV contenant les paramètres de l'instruction `init_params` est également créé. Les fichiers sont nommés d'après le nom de l'image.

La touche Escape permet de quitter le programme.

### 7.3. La procédure `init_sub`

Cette procédure initialise le paramètre  $c$ , la variable  $z$  et la dérivée  $dz$  (dérivée par rapport à  $c$ ). Les formules sont différentes selon l'ensemble choisi (Mandelbrot ou Julia).

Remarquez l'utilisation des nombres complexes, signalés par le suffixe @.

### 7.4. La procédure `iter_sub`

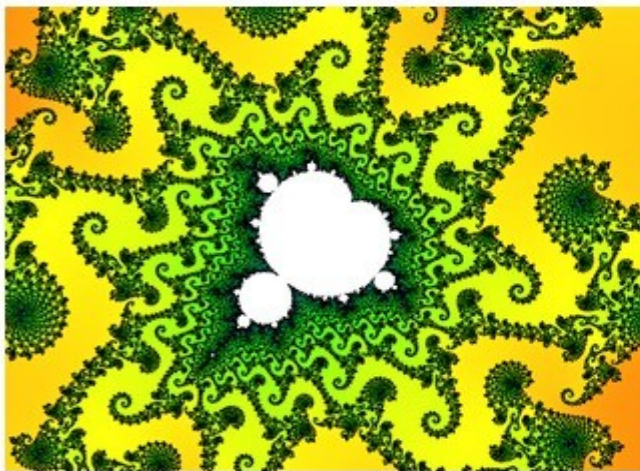
Cette procédure applique les formules permettant de passer de l'itération  $n - 1$  ( $z$ ,  $dz$ ) à l'itération  $n$  ( $zn$ ,  $dzn$ ). Il est nécessaire de calculer à la fois la fonction complexe et sa dérivée par rapport à  $c$ . Là aussi le calcul se fait directement sur les nombres complexes.

## 8. Exemples

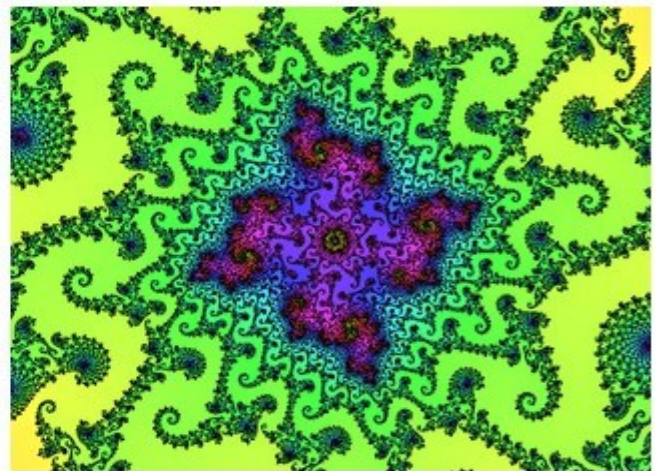
Nous présentons ici quelques images supplémentaires tirées des régions les plus caractéristiques de l'ensemble de Mandelbrot.

### 8.1. La vallée des hippocampes

Nous avons déjà rencontré cette région dont le versant « est » contient des structures en forme d'hippocampe, tandis que le versant « ouest » contient des structures en forme de double spirale. Ici, un mini-ensemble de Mandelbrot et le « noyau » d'un EJS situé près de ce mini-ensemble :



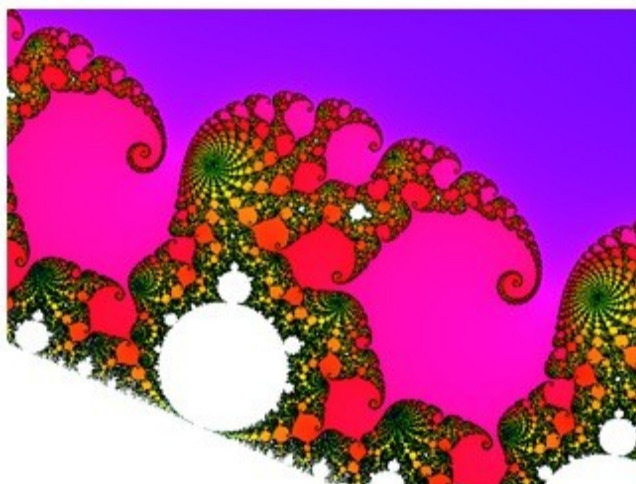
seahorse\_05



seahorse\_07

### 8.2. La vallée des éléphants

Cette région, qui correspond au creux de la cardioïde, révèle des motifs énormément trompeurs :)



elephant\_01



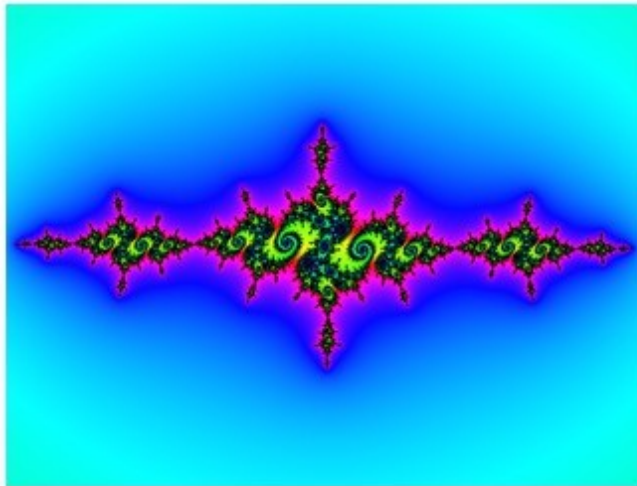
elephant\_05



### 8.3. La vallée des sceptres

Cette région, située entre le disque principal et son « bourgeon » ouest, montre des structures comparables à celles de la vallée des hippocampes, mais hérissées de « décorations » en forme de sceptres.

Ici, un ensemble de Julia et l'agrandissement de sa partie centrale :



julia\_scepter\_01



julia\_scepter\_02

## 9. Conclusion

Dans ce premier article consacré aux fractales, nous avons présenté les ensembles de Mandelbrot et de Julia obtenus avec la fonction classique :  $f(z) = z^2 + c$

Le prochain article examinera le cas  $f(z) = z^p + c$  où  $p$  est un exposant pouvant différer de 2.

## 10. Liens

- [Images des mathématiques](#). Site du CNRS. Nombreux articles sur les fractales et sujets apparentés.
- [Mu-Ency](#). *Encyclopedia of Mandelbrot set*, par Robert Munafo. Décrit notamment la méthode de coloration utilisée dans PANOMAND.
- [High Precision Deep Zoom](#). Images de l'ensemble à très forts agrandissements.
- [Fractal Forums](#). Un ensemble de forums recouvrant tous les aspects des fractales, avec de superbes images.
- [Element90](#). Un blog sur les fractales.