

Devoir maison n° 10

Pour lundi 7 janvier

Un nombre complexe $c \in \mathbb{C}$ étant fixé, on définit une suite complexe (z_n) par :

$$\begin{cases} z_0 = 0 \\ \forall n \in \mathbb{N}, z_{n+1} = z_n^2 + c \end{cases}$$

Le but du problème est de comprendre (au moins dans une certaine mesure) le comportement de la suite (z_n) , en fonction du paramètre c choisi.

Propriétés élémentaires

1. Expliciter la suite (z_n) lorsqu'on prend :

a) $c = 0$

b) $c = -2$

c) $c = -1$

d) $c = i$

2. Montrer que si $c = 1$, alors (z_n) est une suite de nombres réels telle que $\forall n \in \mathbb{N}, z_n \geq n$.

3. Dans cette question, on suppose qu'il existe un rang $N \in \mathbb{N}$ tel que $|z_N| > 2$.

a) On note α la racine positive de l'équation $x^2 - x - |c| = 0$. Montrer que $\alpha \geq 1$.

b) Pour tout entier $n \in \mathbb{N}$, on pose $x_n = |z_n| - \alpha$.

Montrer que $\alpha + x_{n+1} \geq (\alpha + x_n)^2 - |c|$ puis que $x_{n+1} \geq 2\alpha x_n$.

c) Dédurre de cette dernière inégalité que (x_n) tend vers $+\infty$.

d) On ne fait plus d'hypothèse sur (z_n) . Dédurre de ce qui précède que, selon la valeur de c , une et une seule des propriétés suivantes est vérifiée :

(1) La suite (z_n) est bornée.

(2) $|z_n| \xrightarrow[n \rightarrow +\infty]{} +\infty$

et que de plus, dans le premier cas, $|z_n|$ est majoré par 2.

On appelle \mathbb{M} l'ensemble des nombres complexes $c \in \mathbb{C}$ pour lesquels la suite (z_n) est bornée¹.

4. Parmi les complexes $0, -2, -1, i, 1, 2i + 1$, lesquels appartiennent à \mathbb{M} ?

5. Montrer que si $|c| > 2$ alors $c \notin \mathbb{M}$.

En déduire que \mathbb{M} est inclus dans le disque de centre 0 et de rayon 2.

6. Montrer que si le nombre complexe c appartient à \mathbb{M} , alors son conjugué \bar{c} appartient aussi à \mathbb{M} .

En déduire que \mathbb{M} possède un axe de symétrie que l'on précisera.

1. \mathbb{M} est appelé l'ensemble de Mandelbrot, d'après Benoît Mandelbrot (1924-2010).

Partie A- L'intersection $\mathbb{M} \cap \mathbb{R}$

On notera dorénavant $P(z) = z^2 + c$, de telle sorte que $z_{n+1} = P(z_n)$ pour tout $n \in \mathbb{N}$.

Dans cette partie uniquement, c désigne un nombre réel.

La suite (z_n) est donc une suite réelle, par récurrence immédiate.

7. On suppose dans cette question que $-2 \leq c \leq 0$. Montrer que $c \leq z_n \leq -c$ pour tout $n \in \mathbb{N}$.

8. On suppose maintenant que $c > 1/4$. Montrer que $z_{n+1} - z_n \geq c - 1/4$ puis que $z_n \rightarrow +\infty$.

9. On suppose ici que $c \in]0; 1/4[$.

a) Montrer que l'équation $P(x) = x$ a une unique racine ℓ dans l'intervalle $]0; 1/2[$.

b) Montrer que $P(]c; \ell[) \subset]c; \ell[$ (L'intervalle $]c; \ell[$ est stable par P).

En déduire que $z_n \in]c; \ell[$ pour tout $n \in \mathbb{N}^*$.

c) Montrer qu'il existe une constante $K \in [0; 1[$ telle que $|P'(x)| \leq K$ sur $]c; \ell[$.

d) Établir la relation $z_{n+1} - \ell = \int_{\ell}^{z_n} P'(x) dx$.

En déduire que $|z_{n+1} - \ell| < K|z_n - \ell|$ pour tout n .

e) En déduire que $|z_n - \ell| \leq K^{n-2}|z_2 - \ell|$ pour $n \geq 2$ et montrer que (z_n) est convergente.

10. Montrer que pour $c = 1/4$ la suite (z_n) est bornée et en déduire finalement que $\mathbb{M} \cap \mathbb{R} = [-2; 1/4]$.

Partie B- Une cardioïde incluse dans \mathbb{M}

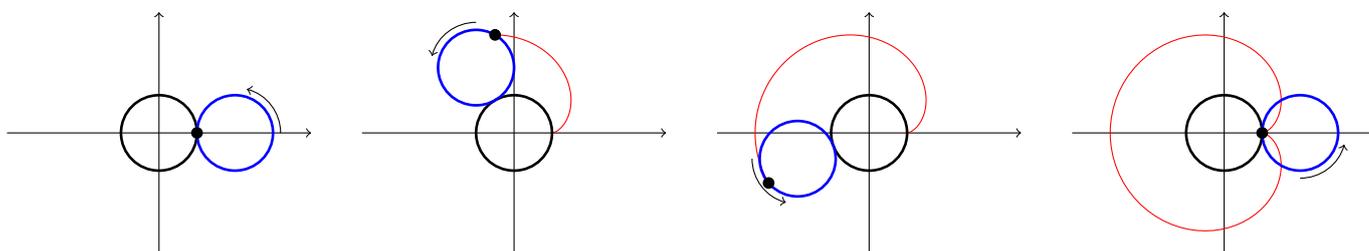
On appelle cardioïde² la partie de \mathbb{C} suivante : $\mathcal{C} = \{\lambda - \lambda^2 \mid \lambda \in \mathbb{C} \text{ et } |\lambda| = 1/2\}$

et cardioïde pleine la partie de \mathbb{C} : $\mathcal{C}^o = \{\lambda - \lambda^2 \mid \lambda \in \mathbb{C} \text{ et } |\lambda| < 1/2\}$.

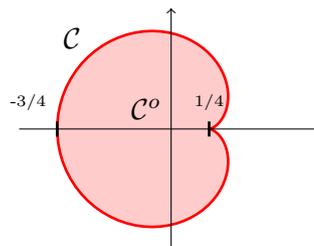
11. (*Allure de la cardioïde*)

a) Montrer que \mathcal{C} est l'ensemble des complexes de la forme $\frac{1}{2}e^{i\theta} + \frac{1}{4}e^{\pi+2i\theta}$, pour $\theta \in \mathbb{R}$.

b) Expliquer pourquoi \mathcal{C} est la courbe décrite par un point de la circonférence d'un cercle de rayon $1/4$ qui roule sans glisser sur un autre cercle, fixe, de centre 0 et de rayon $1/4$.



Allure des ensembles \mathcal{C} et \mathcal{C}^o :



2. « en forme de cœur »

À partir de maintenant, nous admettons le résultat suivant :

Si $Q = P^n = P \circ P \circ \dots \circ P$ est le polynôme P composé n fois ($n \in \mathbb{N}^*$), si $\ell \in \mathbb{C}$ est un point fixe de Q , et si la dérivée du polynôme Q en ℓ a un module strictement inférieur à 1 ($|Q'(\ell)| < 1$), alors la suite (u_n) définie par $u_0 = 0$ et $\forall n \in \mathbb{N}, u_{n+1} = Q(u_n)$ converge vers ℓ .

Ce résultat est une généralisation de ce qu'on a fait à la question 8, basée sur la même idée (« si la dérivée est plus petite que 1 autour du point fixe, les points de la suite ont tendance à s'en rapprocher »). La démonstration de ce résultat est un peu délicate, et nous la laissons de côté.

12. (\mathcal{C}^o est inclus dans \mathbb{M})

Soit $c \in \mathcal{C}^o$. Donner un point fixe de P qui s'exprime simplement.

Montrer que $c \in \mathbb{M}$.

Partie C- Un disque inclus dans \mathbb{M}

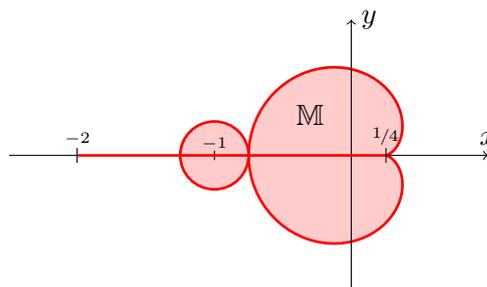
On note $\mathcal{D}^o = \{z \in \mathbb{C} \mid |z + 1| < 1/4\}$ le disque ouvert de centre -1 et de rayon $1/4$.

Dans toute cette partie, on prend $c \in \mathcal{D}^o$.

13. Montrer que c peut s'écrire $c = -1 - \lambda - \lambda^2$ où $\lambda \in \mathbb{C}$ vérifie $|\lambda^2 + \lambda| < 1/4$.
14. Montrer que la suite (z_{2n}) vérifie une relation de récurrence $z_{2n+2} = Q(z_{2n})$ où Q est un polynôme que l'on exprimera à l'aide de P .
15. Montrer que λ est un point fixe de Q et que $|Q'(\lambda)| < 1$.
16. En déduire que (z_{2n}) converge vers λ .
17. Quel est la nature de la suite (z_{2n+1}) ?
18. Déduire des questions précédentes que $c \in \mathbb{M}$.
19. Est-ce que la suite (z_n) converge ?

D- Tracé informatique

À ce stade de l'étude, on sait que \mathbb{M} est inclus dans le disque de centre 0 de rayon 2, qu'il contient le segment $[-2; 1/4]$ ainsi que la cardioïde et le disque des parties précédentes.



On souhaite tracer \mathbb{M} sur l'ordinateur pour voir les détails qui nous échappent. Pour ce faire, on va ouvrir une fenêtre de taille 640×480 et on va colorer chaque pixel de cette fenêtre en blanc (s'il n'est pas dans \mathbb{M}) ou en noir (s'il est dans \mathbb{M}).

Pour tester si un point c appartient à \mathbb{M} , le principe est le suivant : on calcule les 150 premiers termes de la suite (z_n) , et si aucun de ces z_n n'a un module dépassant 2, le point c sera réputé appartenir à \mathbb{M} . (Bien entendu il s'agit d'une approximation. Ce choix arbitraire de 150 itérations peut être modifié. En l'augmentant on obtient un tracé plus précis, au prix d'un temps de calcul plus long). Le code utilisé se trouve sur la page suivante.

```

from PIL import Image # Pour manipuler des images

NMAX = 150 # Nombre maximal d'itérations
LARGEUR, HAUTEUR = 640, 480 # Taille de la fenêtre
YMIN, YMAX, XMIN = -1.25, 1.25, -2.4 # Coordonnées d'affichage
XMAX = XMIN+(YMAX-YMIN)*LARGEUR/HAUTEUR

# Alloue l'espace pour l'image de sortie
img = Image.new('RGB', (LARGEUR, HAUTEUR))
pixels = img.load()

def test(c):
    # à compléter

for x in range(LARGEUR):
    re = (x/LARGEUR) * (XMAX-XMIN) + XMIN
    for y in range(HAUTEUR):
        im = (y/HAUTEUR) * (YMAX-YMIN) + YMIN
        nombre_iters = test( re + 1J*im )
        if nombre_iters == NMAX:
            pixels[x,y] = (0,0,0) # pixel noir
        else:
            pixels[x,y] = (255,255,255) # pixel blanc

img.show() # Affiche le résultat
img.save("mandelbrot.png", "png") # Enregistre dans un fichier

```

20. Entrer le programme et le compléter en écrivant la fonction `test(c)` qui prend un complexe c et renvoie le premier entier n tel que $|z_n| > 2$, ou `NMAX` si z_{NMAX} est encore de module inférieur à 2. Puis exécuter le programme.

21. Le tracé précédent n'est pas entièrement satisfaisant parce qu'on ne voit pas tous les détails de la frontière de \mathbb{M} : ces détails sont trop fins par rapport à la grille de pixels.

Pour mieux visualiser la frontière de \mathbb{M} , on peut rajouter un dégradé de couleur à l'extérieur de l'ensemble \mathbb{M} . Remplacer la ligne :

```
pixels[x,y] = (255,255,255) # pixel blanc
```

par les lignes suivantes (en n'oubliant pas d'importer la fonction `sqrt`) :

```
code_couleur = int(sqrt(nombre_iters)*20.8)
pixels[x,y] = (code_couleur, code_couleur, 255)
```

On peut ensuite expérimenter et changer les coordonnées de la fenêtre d'affichage, le nombre maximal d'itérations, le choix des couleurs, etc...