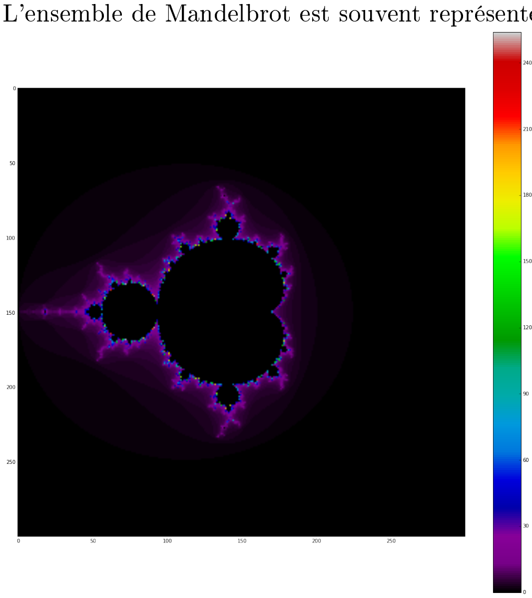
L3 MIASHS – Informatique S5
TD4 : Mandelbrot

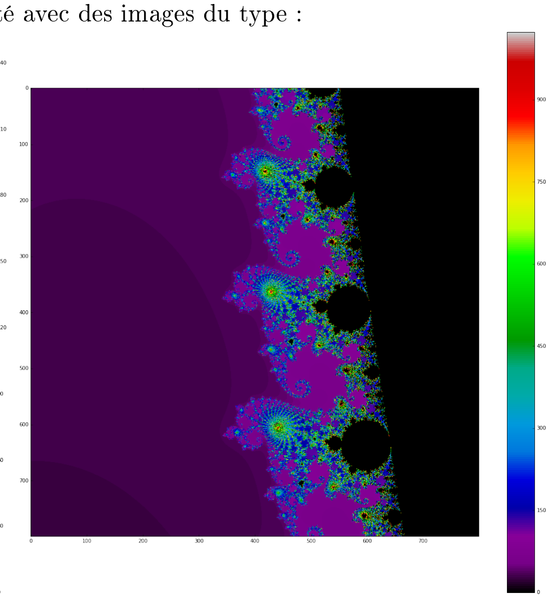
2019-2020

Sujet très très fortement inspiré de : yaspat.github.io/blog/posts/utilisation-de-numpy-ensemble-de-mandelbrot.

1 Ensemble de Mandelbrot

L'ensemble de Mandelbrot est souvent représenté avec des images du type :





L'objectif est de construire de telles images.

Soit la suite complexe (z_n)définie par récurrence par :

$$\forall n \geq 0, z_{n+1} = z_n^2 + z_0$$

Pour un point de départ z_0, on s'intéresse au premier indice i tel que $|z_i| > 2$. Ce i est appelé *durée de vie* de z_0.

Les images sont obtenues en associant aux points du plan complexe une couleur qui représente la durée de vie de chaque point.

2 Calculer la durée de vie

2.1

Construire une fonction `liste_suite(z_0,n)` qui prend un complexe z_0 et un entier n et retourne en sortie la liste `[z0,z1,...,zn]`.

Calculer les listes `liste_suite(z0,10)` pour `z0` dans $\{0, 1, -1, i, -i, \frac{1+i}{2}\}$.

Remarque : on note `j` le i des complexes et par exemple `0.5+0.5j` pour $\frac{1+i}{2}$.

.

2.2

Construire la fonction `duree_de_vie(z0)` qui calcule la durée de vie d'un complexe `z0`.

On utilise `abs` aussi pour calculer le module des nombres complexes.

Précision : Par convention, si on atteint le 256ème terme de la suite sans sortir du disque de rayon 2, on décide que la durée de vie est fixée arbitrairement à 0.

Vérification : Calculez la durée de vie pour chacun des z_0 pris en exemple à la question précédente. Vérifiez en affichant la liste des premiers termes pour certains d'entre eux que votre fonction donne le bon résultat.

3 Systématiser le calcul et fabriquer l'image

3.1

On va donc colorier les points du plan du complexe avec une couleur qui corresponde à sa durée de vie. Les premières images du TP illustre ce principe avec la légende des couleurs à droite.

On montre facilement que si un terme de la suite (z_n) dépasse 2 en module, alors c'est le cas de tous les suivants. Donc, si $|z_0| > 2$, la durée de vie de ce point ne nous intéresse pas, elle est de 0.

On va donc se limiter aux points du plan complexe dont les parties rélle et imaginaire sont comprises entre -2 et 2 :

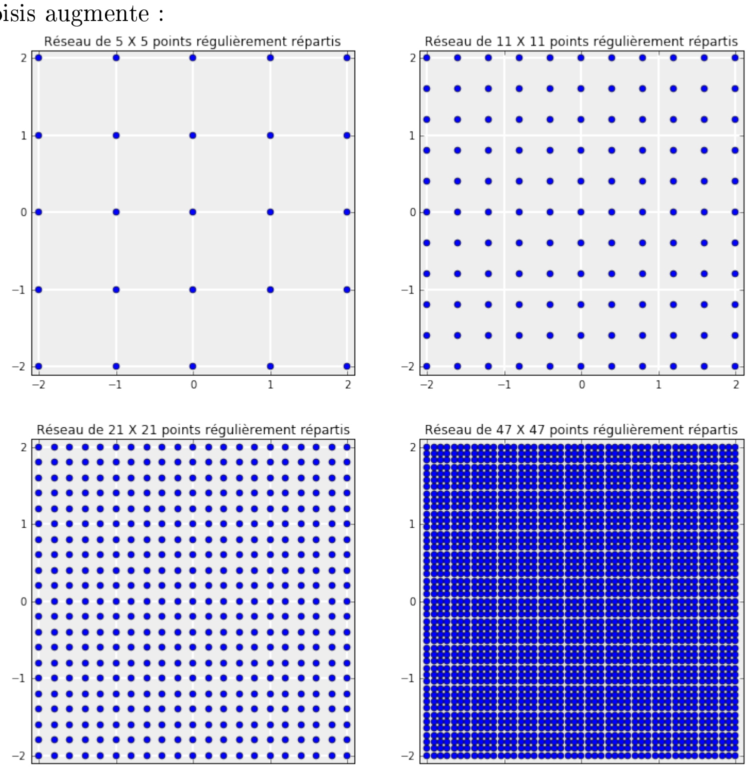
$$R = \{x + iy, -2 \leq x \leq 2, -2 \leq y \leq 2\}$$

L'ensemble R est donc un carré.

Comme l'ensemble R contient une infinité de points, on ne peut pas calculer la durée de vie de tous les points qu'il contient. Il faut donc en sélectionner un nombre suffisant pour avoir une image précise.

Pour cela on va définir un maillage de $p * p = p^2$ le nombre de points répartis régulièrement dans le carré R. Et ce sont ces points dont on va ensuite calculer la durée de vie.

Voici à quoi on veut que le maillage ressemble au fur et à mesure que le nombre de points choisis augmente :



Le code suivant qui permet de calculer un tel ensemble de points :

```python
def reseau(nb_points):
    # construction des abscisses
    x = np.linspace(-2,2,nb_points) # nb_points regulierement
                                    # repartis de (-2) a 2
    # Construction des ordonnees
    y = np.linspace(2,-2,nb_points) # Attention les ordonnees vont decroissant
    X,Y = np.meshgrid(x,y) # J'ai maille mon domaine
                           # meshgrid retrourne deux matrices
    return X+1j*Y
```

Vérifiez qu'il fonctionne, en affichant les valeurs des points pour `nb_points` valant 5, 10, puis leur représentation avec `pyplot` comme dans l'image précédente, avec les valeurs 5, 10 et 100.

3.2

Ecrivez une fonction qui prend un array de points et retourne un array contenant les durées de vie de chacun des points. Pensez à utiliser votre fonction `liste_suite`.

3.3

Ecrivez une fonction qui prend un nombre de points et affiche la fractale de Mandelbrot pour ce nombre de points. Utilisez la question précédente.

Pour afficher le résultat, cherchez la commande `plt.imshow(image, cmap = 'spectral')`.

3.4

Regardez la fonction proposée là :
http://yaspat.github.io/blog/posts/utilisation-de-numpy-ensemble-de-mandelbrot/#trace-de-l-image

Quelles différences ? Quel intérêt ?

Utilisez cette solution.