

Commentaires concernant le programme p. 23 du cours

Ouvrir un logiciel en ligne pour programmer en Python, par exemple le site : <https://www.gleven.fr/python/>

Ecrire le script du programme dans l'éditeur puis, dans la console le logiciel le traduit en langage machine (une suite de 0 et de 1) et on exécute le programme.

La bibliothèque (que l'on doit appelée avec « from » suivi du nom de la bibliothèque) utilisée pour tracer des courbes est « matplotlib.pyplot » que l'on nome plus simplement dans le script (avec « as » suivi du nouveau nom) en ayant défini un nouveau nom (le plus classique est « plt »).

Pour ajouter un élément à la liste il faut mettre le nom de la liste puis « .append() » et mettre dans la parenthèse la valeur à ajouter. Ici en mettant « x.append(float(x[k]) + 1/N) », j'ai ajouté la valeur « x_{k+1} » à la liste en utilisant la formule de récurrence donnée dans l'exercice : « x[k] » prend la valeur placée à la (k+1)^{ème} position dans la liste (la numérotation commençant toujours à 0 : voir le II de la couverture du livre pour les instructions concernant les listes).

Il est mieux de préciser la nature du nombre x_k qui prend des valeurs avec un grand nombre de chiffres : ici on a un réel d'où « float() », pour entier naturel il faudrait mettre « int() » et pour une chaîne de caractères on mettra entre guillemets (simples ou doubles).

Pour faciliter la lecture on peut définir la valeur à rentrer dans la liste avant de la mettre dedans (c'est le cas de b et y) et donc on aurait pu écrire « a = float(x[k]) + 1/N » puis « x.append(a) » au lieu de « x.append(float(x[k]) + 1/N) ».

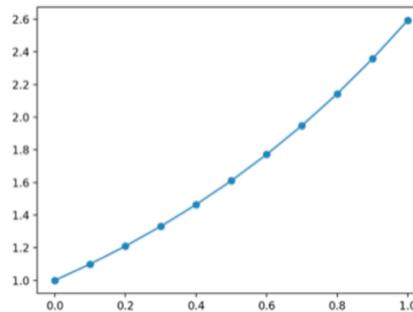
On pouvait aussi concasser des listes en ajoutant la liste contenant la nouvelle valeur à la liste existante : « x+[a] » ajoute la liste [a] à la liste x (voir le tutoriel sur les listes mis dans l'annexe cours - chapitre 9).

Nouveau script possible :

```
from matplotlib import pyplot as plt
N = 10
x = [] #liste des abscisses
y = [] #liste des ordonnées
x = x + [0]
y = y + [1]

for k in range(N):
    a = float(x[k]) + 1/N
    x = x + [a]
    b = (1 + 1/N)*float(y[k])
    y = y + [b]
    # Affichage des points dans le repère

plt.plot(x, y, marker='o', linestyle='-')
plt.show()
```



« plt.plot(x, y, 'r+') » dessine les points en prenant à chaque fois la valeur dans la liste x pour l'abscisse du point, la valeur dans la liste y pour l'ordonnée du point et « 'r+' » donne la couleur et la forme du point : vous pouvez changer la couleur et la forme des points en remplaçant « plt.plot(x, y, 'r+') » par exemple par « plt.plot(x, y, 'b-') » En rajoutant `linestyle='-'` on relie les points par des traits.