

Les documents de cours sont autorisés.

Questions :

1) A quoi sert le symbole # ?

.....

.....

2) A quoi sert le symbole = ?

.....

.....

3) A quoi sert l'instruction *print* ?

.....

.....

4) Quelle est la signification de l'instruction suivante :
 $a, b = b, a$

.....

.....

5) On considère l'algorithme suivant :

L1	Traitement :	$U \leftarrow 20$
L2		$N \leftarrow 0$
L3		$U \leftarrow 0,9 \times U + 4$
L4		$N \leftarrow N + 1$

Ecrire en langage Python ce programme en utilisant le moins de lignes possibles

.....

.....

.....

.....

6) Quelle instruction permet de définir une fonction ?

.....

7) Définir sur votre console, en langage Python une fonction g de la variable t qui à t associe $g(t) = 2t^3 - 3t^2 + 7t - 4$

8) On rappelle que l'instruction `range(début, fin, pas)` retourne une liste de nombres entiers. Les arguments `fin` et `pas` sont optionnels.

Créer une fonction `tableau(f, a, b)` renvoyant la liste des images des entiers de a à b par une fonction f .

Appliquer cette fonction à la fonction g définie précédemment afin d'obtenir le tableau valeurs de la fonction g .

.....

.....

.....

.....

.....

.....

.....

.....

9) Tracer à l'écran le nuage de points issu du tableau de valeurs crée précédemment en modifiant éventuellement le script envoyé par le professeur. Appeler le professeur quand cela est réalisé.

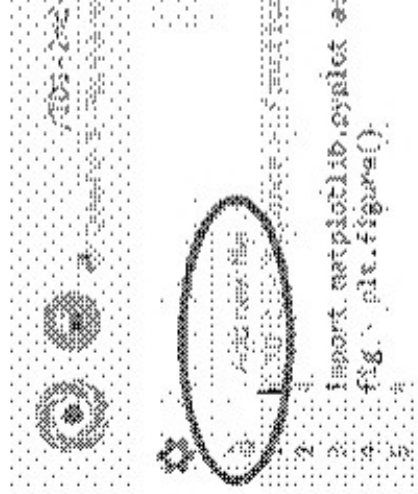
On peut facilement tracer les nuages de points correspondants, en utilisant les fonctions du module `matplotlib`. On importe le sous-module `pyplot` de `matplotlib` qu'on renomme au passage `plt`, par commodité.

Sur un *python* installé, pas de problème par contre sur *repl* ou les logiciels *online*, le graphique doit être converti en image. Sur *repl*, il faut en outre convertir avec `Pillow` avec `new fig` en haut à gauche.

Après avoir écrit la fin de votre programme : Cliquer sur `Add new file`, puis sur `Continue` et enfin sur `run ...`. Admirez le résultat !

Code Python

```
import matplotlib.pyplot as plt
fig = plt.figure() # nécessaire seulement sur repl
(x, y) = tableau valeurs(f, 0, 10)
plt.plot(x, y, '.', color='red') # la 3e argument : permet de
préciser qu'on veut un nuage de points le 4e est explicite ...
fig.savefig('graph.png') # ou simplement plt.savefig sur un ip
# pour installer comme auconda
```



Remarque sur `plt.plot(x, y, '.', color='red')`

Si on omet le troisième argument, `matplotlib` va relier deux points consécutifs par un segment de droite.