# Suites arithmétiques

[**https://www.mathweb.fr/euclide/suites-arithmetiques-et-geometriques-avec-python/**](https://www.mathweb.fr/euclide/suites-arithmetiques-et-geometriques-avec-python/)

**Première méthode : liste des premiers termes**

Pour calculer les premiers termes d’une suite arithmétique, on peut alors définir une fonction comme cela :

def suite\_arithmetique (terme , raison , indice\_final):

U = [ terme ]

for n in range(indice\_final):

terme += raison

U.append( terme )

return U

Expliquons cette fonction :

* Pour commencer, la fonction se nomme “suite\_arithmetique” et possède 3 arguments : le premier est nommé “terme” (il va correspondre au premier terme de la suite que l’on connait), le deuxième est nommé “raison” et correspond à la raison de la suite, et le dernier est nommé “indice\_final” et correspond à l’indice du dernier terme que l’on souhaite connaître.
* On commence par définit une liste nommée “U” qui contient le premier terme de la suite (ligne 2), donc *u*0.
* Ensuite, on créée une boucle “for” comportant “indice\_final” itérations car il faudra calculer *u*1, *u*2, … , *un* (il y a bien *n* termes à calculer).  
  Dans cette boucle, on ajoute au terme connu la raison (ligne 4), puis on l’insère dans la liste (avec la méthode “append”, ligne 5).
* Une fois la boucle terminée, la fonction retourne la liste U obtenue, qui contient alors tous les termes.

def suite\_arithmetique(terme , raison , indice\_final):

U = [ terme ]

for n in range(indice\_final):

terme += raison

U.append( terme )

return U

print (suite\_arithmetique(1,2,10))

Cela donne par exemple, pour la suite arithmétique de premier terme *u*0=3 et de raison *r* = 5 :

>>> suite\_arithmetique(3,5,20)

[3, 8, 13, 18, 23, 28, 33, 38, 43, 48, 53, 58, 63, 68, 73, 78, 83, 88, 93, 98, 103]

#### Autre méthode : affichage des premiers termes

Si on ne souhaite pas avoir de liste, mais un simple affichage, on peut utiliser une méthode plus directe, comme celle-ci :

n = 0

u = 3

r = 5

print(f'u({n}) = {u}')

for n in range(20):

u = u + r

print(f'u({n+1}) = {u}')

qui affiche :

u(0) = 3

u(1) = 8

u(2) = 13

u(3) = 18

u(4) = 23

u(5) = 28

u(6) = 33

u(7) = 38

u(8) = 43

u(9) = 48

u(10) = 53

u(11) = 58

u(12) = 63

u(13) = 68

u(14) = 73

u(15) = 78

u(16) = 83

u(17) = 88

u(18) = 93

u(19) = 98

u(20) = 103

J’ai ici utilisé un formatage pour plus de lisibilité (voir [page sur les formatages](https://www.mathweb.fr/euclide/le-formatage-dune-chaine-de-caracteres-en-python/)).