

```
s = experience()
n = 1
L = [s] # moyenne su
while n < nExperiences:
    n = n+1
    s = s + experience
    L.append(s/n) # o
plt.plot(list(range(1,
plt.plot([1, nExperiend
```

# RECHERCHE DE NOMBRES PREMIERS

1) L'algorithme ci-dessous doit permettre de vérifier si un nombre est premier :

Python
<pre>from math import* def premier(n):     print(n)     p=2     while [ ]:         if [ ]:             print("est divisible par",p)             break         p=p+1</pre>

TI	CASIO
<pre>:Input N :Disp N :2→P :While [ ] :If [ ] :Then :Disp "EST DIVIS IBLE PAR",P :Stop :End :P+1→P :End</pre>	<pre>"N="?→Ne N, 2→P While [ ] If [ ] Then "EST DIVISIBLE P AR", P, Stop IfEnd P+1→P WhileEnd</pre>

- Compléter les instructions cachées dans ce programme.
- Qu'affiche le programme en sortie si le nombre entré est premier ?
- Tester ce programme sur un ordinateur ou une calculatrice pour trouver quelques nombres premiers supérieurs à 1000.

2) Le mathématicien suisse *Leonhard Euler* (1707 - 1783) a découvert en 1772 une formule simple permettant de produire de nombreux nombres premiers :

$$n^2 + n + 41$$

Par exemple, pour  $n = 0$ , la formule renvoie 41 qui est premier.

- Vérifier, pour d'autres valeurs entières de  $n$ , si le nombre renvoyé par la formule est premier.
- Modifier le programme précédent pour tester si la formule d'Euler renvoie un nombre premier pour tout  $n$  compris entre 0 et 100.
- Jusqu'à quelle valeur de  $n$  la formule renvoie-t-elle systématiquement un nombre premier.
- Trouver quelques nombres premiers supérieurs à un million.



Hors du cadre de la classe, aucune reproduction, même partielle, autres que celles prévues à l'article L 122-5 du code de la propriété intellectuelle, ne peut être faite de ce site sans l'autorisation expresse de l'auteur.

[www.maths-et-tiques.fr/index.php/mentions-legales](http://www.maths-et-tiques.fr/index.php/mentions-legales)