Arithmétique : nombres premiers
Problème de recherche : le polynôme d'Euler

En 1772, Leonhard Euler signale que le polynôme $P(n) = n^2 + n + 41$ prend des valeurs premières pour de nombreux entiers n consécutifs.

La mission

1. Écrire une fonction Python `premier(N)` qui renvoie `True` si N est premier. Déterminer les entiers n de 0 à 50 pour lesquels $P(n)$ n'est pas premier.
2. Démontrer sans calculatrice que $P(40)$ et $P(41)$ ne sont pas premiers.
3. Démontrer que pour tout entier naturel k, $P(41k)$ est divisible par 41. En déduire qu'il existe une infinité d'entiers n pour lesquels $P(n)$ n'est pas premier.
4. Soit $Q(n) = an^2+bn+c$ avec a, b, c entiers et $a \neq 0$. On suppose que $Q(n_0) = p$ est premier pour un certain entier n_0. Démontrer que $Q(n_0+kp) \equiv 0 \pmod{p}$ pour tout entier k. En déduire qu'un tel polynôme ne peut pas donner uniquement des nombres premiers pour $n \in \mathbb{N}$.

Pour aller plus loin

Vérifier avec Python que $R(n) = n^2 - 79n + 1601$ est premier pour tous les entiers de 0 à 79. Démontrer que $R(n) = P(n-40)$ et expliquer ce résultat.

Coups de pouce

Bloqué ? Des indices progressifs sont disponibles en ligne. N'en ouvrir qu'un à la fois, et seulement après avoir cherché !