Python
Problème de recherche : les nombres heureux

On part d'un entier strictement positif et on le remplace par la somme des carrés de ses chiffres, puis on recommence. Par exemple :

$$7 \to 49 \to 97 \to 130 \to 10 \to 1 \to 1 \to \cdots \qquad (49 \to 4^2 + 9^2 = 97).$$

Un nombre est dit **heureux** si l'on finit par atteindre 1. Sinon, il est dit **malheureux**.

La mission

1. Quels sont les nombres heureux entre 1 et 100 ? Combien y en a-t-il ?
2. Que se passe-t-il pour les nombres malheureux ? Formuler une conjecture.
3. Pourquoi la suite des nombres obtenus ne peut-elle pas grandir indéfiniment ?

Pour aller plus loin

On remplace maintenant un nombre par la somme des *cubes* de ses chiffres. Trouver tous les nombres de 2 à 999 qui sont égaux à la somme des cubes de leurs chiffres (par exemple $153 = 1^3 + 5^3 + 3^3$).

Coups de pouce

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

Éléments de correction

Un programme possible

```
def somme_carres(n):
    s = 0
    while n > 0:
        s = s + (n % 10)**2
        n = n // 10
    return s
def est_heureux(n):
    while n != 1 and n != 4:
        n = somme_carres(n)
    return n == 1

print([n for n in range(1, 101) if est_heureux(n)])
```

1. Les nombres heureux jusqu'à 100

Il y en a **20** : 1, 7, 10, 13, 19, 23, 28, 31, 32, 44, 49, 68, 70, 79, 82, 86, 91, 94, 97, 100.

2. Les nombres malheureux

En affichant les nombres successifs, on constate qu'ils finissent tous par tomber dans le même cycle :

$$4 \to 16 \to 37 \to 58 \to 89 \to 145 \to 42 \to 20 \to 4.$$

Conjecture : tout nombre malheureux finit dans ce cycle. (C'est pourquoi le programme s'arrête quand il atteint 4.)

3. La suite ne grandit pas indéfiniment

Un nombre de k chiffres ($k \geqslant 4$) est au moins égal à 10^{k-1}, alors que la somme des carrés de ses chiffres vaut au plus $81k$. Pour $k = 4$: $81 \times 4 = 324 < 1\,000$, et l'écart ne fait que grandir ensuite. Donc, dès qu'un nombre a 4 chiffres ou plus, le suivant est plus petit. Les nombres finissent par être inférieurs à 1 000, puis ils restent inférieurs ou égaux à $3 \times 81 = 243$. Il n'y a qu'un nombre fini de valeurs possibles : la suite finit forcément par repasser par une valeur déjà vue, et donc par tourner en boucle. Un programme qui teste tous les nombres de 1 à 243 montre que les seules boucles sont $1 \to 1$ et le cycle de 4 : la conjecture est démontrée (avec l'aide de l'ordinateur).

Pour aller plus loin

En modifiant la fonction (puissance 3 au lieu de 2) et en testant tous les nombres, on trouve **153, 370, 371 et 407**.

Ce qu'on a utilisé du chapitre

- Fonctions, boucles `while` et `for`, conditions ; quotient et reste (`//` et `%`).
- Utilisation d'un programme pour explorer, conjecturer et vérifier un grand nombre de cas.