

## Travaux pratiques et approche expérimentale

- **Obtention de PGCD et coefficients de Bezout de deux entiers naturels a et b**

Le **PGCD** des deux entiers est obtenu par l'**algorithme d'Euclide** : boucle réalisée tant que le reste est non nul.

Application à 3018708 et 481338 :  
 $3018708 \wedge 481338 = 2178$ .

Les **coefficients de Bezout**, a et b entiers sont tels que  $a \times x + b \times y = x \wedge y$ . Ils sont obtenus par **récurtivité** (Bezout(..) étant présente dans sa propre définition.)

Application à 3018708 et 481338 :  
 $70 \times 3018708 - 439 \times 481338 = 2178$

```
1 def PGCD(x, y):
2     chaine = {}
3     while x % y != 0:
4         x, y = y, x % y
5     return str(y)
6
7 def Bezout(x, y):
8     if y == 0:
9         return (1, 0)
10    else:
11        (q, r) = divmod(x, y)
12        (w, z) = Bezout(y, r)
13        return (z, w - q * z)
14
15 print PGCD(3018708, 481338)
16 print Bezout(3018708, 481338)
17
18 PGCD = 2178
19 (a, b) = (70, -439)
```

- **Obtention de la décomposition en facteurs premiers d'un entier naturel n ≥ 1**

Tout **entier naturel n ≥ 1** s'écrit de façon **unique** comme **produit de puissances** ( $\delta_i$ , calculées avec `chaine[di]`) de **nombre premiers** ( $p_i$ , calculés avec `k`) :

$$n = p_1^{\delta_1} p_2^{\delta_2} \dots p_m^{\delta_m}.$$

Application à  $m = 481338$  :  
 $m = 2^1 \times 3^2 \times 11^2 \times 13 \times 17$ .

Application à  $n = 3018708$  :  
 $n = 2^2 \times 3^4 \times 7^1 \times 11^3$ .

On retrouve ainsi le **PGCD**,  $m \wedge n$  de ces deux entiers.

```
1 def decomposition(n):
2     chaine = {}
3     dec = 1
4     x = n
5     di = 2
6     while dec < n:
7         while x % di == 0:
8             if di not in chaine:
9                 chaine[di] = 1
10            else:
11                chaine[di] += 1
12            dec *= di
13            x //= di
14            di += 1
15     resultat = str(n) + ' = '
16     for k in sorted(chaine):
17         resultat += str(k) + '^'
18         + str(chaine[k]) + ' * '
19     return resultat
20
21 print decomposition(3018708)
22
23 3018708 = 2^2.3^4.7^1.11^3.
```