TP 9 - Arithmétique dans $\mathbb{N}$

Exercice 1 *Division euclidienne dans $\mathbb{N}$* Donner une fonction `DE(a,b)` qui retourne le couple `(q,r)` tel que :

$$a = qb + r \text{ et } 0 \leq r < b$$

La fonction utilisera uniquement des additions (ou soustractions) comme dans l'algorithme ci-dessous :

Entrées : $a, b \in \mathbb{N}$ avec $b > 0$

$B \leftarrow b$
$R \leftarrow a$
$Q \leftarrow 0$
Tant que $R \geq B$ faire
    $R \leftarrow R - B$
    $Q \leftarrow Q + 1$
FinTantQue

Sortie : (Q, R) le couple quotient-reste

⇛ Pour la suite, on utilisera les instructions PYTHON suivante concernant la division avec quotient entier :

$$a = qb + r \text{ avec } 0 \leq |r| < |b| \text{ et } rb \geq 0$$

Le reste et le diviseur sont du même signe.

Notation	Explication
%	Reste
//	Quotient

```
>>> (18//7,18%7)
(2, 4)
>>> (18//-7,18%-7)
(-3, -3)
```

Exercice 2 *Algorithme d'Euclide* **Prog**

1. Compléter l'algorithme d'Euclide ci-dessous et écrire une fonction `pgcd(a,b)` qui retourne $a \wedge b$.

Entrées : $a, b \in \mathbb{N}$ avec $b > 0$

$A \leftarrow a$
$B \leftarrow b$
Tant que ... faire
    $R \leftarrow \ldots$
    $A \leftarrow \ldots$
    $B \leftarrow \ldots$
FinTantQue

Sortie : A le PGCD de $a$et b

2. Modifier la fonction `pgcd(a,b)` en une fonction `pgcd_details(a,b)` qui affiche les différentes étapes de l'algorithme. Par exemple :

```
>>> pgcd_details(35,21)
35 = 1 * 21 + 14
21 = 1 * 14 + 7
14 = 2 * 7 + 0
```

Exercice 3 *Relation de Bezout* **Prog**

On rappelle une des suites récurrentes associées à l'algorithme d'**Euclide étendu** :

$$\begin{aligned} a = r_0 &= u_0 a + v_0 b \\ b = r_1 &= u_1 a + v_1 b \\ \forall n \geq 0 \quad r_{n+2} &= r_n - q_{n+2} r_{n+1} \end{aligned}$$

Ce qui donne :

$$\begin{cases} u_0 = v_1 = 1 \\ v_0 = u_1 = 0 \end{cases} \quad \text{et } \forall n \geq 0 \quad \begin{cases} u_{n+2} = u_n - q_{n+2} u_{n+1} \\ v_{n+2} = v_n - q_{n+2} v_{n+1} \end{cases}$$

1. Écrire une fonction `Bezout(a,b)` qui retourne le couple de Bezout associé à (a, b).
2. Écrire une fonction `chinois(a,b,p,q)` qui calcule les solutions de :

$$\begin{cases} n \equiv a \ [p] \\ n \equiv b \ [q] \end{cases}$$

où p et q sont deux entiers naturels premiers entre eux.
La fonction retourne une solution particulière et une solution du système homogène associé.

Exercice 4 *Décomposition en facteurs premiers* On rappelle que tout nombre $n \geq 2$ peut s'écrire (de manière unique) sous la forme :

$$n = \prod_{p \in \mathbb{P}} p^{\alpha_p}$$

où $(\alpha_p)_{p \in_b P}$ est un famille d'entiers naturels *presque nulle*.

Entrées : $n \in \mathbb{N}$

L une liste vide
$p \leftarrow 2$
$N \leftarrow n$
Tant que $p^2 \leq N$ faire
    $\alpha \leftarrow$ la valuation de p dans N
    Si $\alpha > 0$ alors ajouter (p, α) à L
    $N \leftarrow \frac{N}{p^\alpha}$
    $p \leftarrow p + 1$
FinTantQue
Si $N > 1$ alors ajouter $(N, 1)$ à L

Sortie : L la décomposition de n

1. Écrire une fonction `valutation(p,n)` qui détermine la valuation p-adique de n.
2. Écrire une fonction `dfp(n)` qui prend un entier (≥ 2) et renvoie sa décomposition en facteurs premiers.

Exercice 5 *Méthode du crible d'Eratosthène* **Prog**

Écrire une fonction `crible(n)` qui déterminer tous les nombres premiers inférieurs à n.
Dans un deuxième temps, chercher à optimiser cette fonction

Lycée Saint-Exupery – maffre.saintexupery@gmail.com – MPSI