DM : implantation de l'algorithme de PGCD rapide

L'objectif de ce DM est d'implanter l'algorithme de PGCD rapide. On utilisera des polynômes à coefficients dans $\mathbb{F}_p$ pour un certain premier p. On représentera les polynômes par une liste de coefficients, le coefficient constant étant le premier de la liste.
Pour manipuler des polynômes à coefficients dans des corps finis, on fournit dans le fichier `corps_fini.py` une implantation de ces corps, qui s'utilise de la manière suivante :

```
>>> from corps_fini import *          # Import des fonctions
>>> K = ZpZ(17)                       # Corps fini à 17 éléments
>>> a, b, c = K(2), K(5), K(7)        # Quelques éléments du corps
>>> a + b*c, a/b, -a                  # Calculs modulo p
(3 (mod 17), 14 (mod 17), 15 (mod 17))
>>> a == c - b                        # Test d'égalité
True
>>> K == a.field()                    # Permet de savoir d'où vient a
True
>>> P = [K(2), K(3), K(4)]            # Polynôme 2 + 3*X + 4*X^2
>>> 2 + a                             # ATTENTION : il faudrait écrire K(2) + a ici !
TypeError: unsupported operand type(s) for +: 'int' and 'ZpZ_elt'
```

Consignes.

- Fournir toutes les implantations demandées, en conservant les noms du sujet. Penser à bien tester systématiquement toutes les fonctions, même dans des cas limites (entrées nulles, etc.). Tester en particulier l'algorithme du PGCD rapide à l'aide de l'algorithme d'Euclide étendu.
- Il est autorisé, voire encouragé, d'écrire d'autres fonctions dont vous avez besoin !
- **Rendu.** Vous devez rendre ce DM avant le 12 décembre 8h45, sous la forme d'un fichier Python contenant toutes les fonctions demandées, déposé à l'emplacement prévu sur Moodle.

Exercice 1. *Opérations de base*

1. Implanter l'algorithme `addition` qui somme deux polynômes. *Attention aux tailles d'entrées.*
2. Implanter l'algorithme `mul_naive` qui effectue le produit de deux polynômes, de manière naïve.
3. Implanter l'algorithme `div_eucl` qui effectue la division euclidienne naïve de deux polynômes.

Exercice 2. *Algorithme du PGCD rapide*

1. Implanter l'algorithme d'Euclide étendu `xGCD`.
2. Implanter l'algorithme `mul_vec` qui prend en entrée une matrice et un vecteur de dimensions 2, contenant des polynômes, et qui effectue le produit matrice-vecteur. *On représente une matrice* 2×2 *par une liste de listes. Chaque entrée de la matrice étant un polynôme, lui-même représenté par une liste, on aura donc des listes de listes de listes.*
3. Implanter de même l'algorithme `mul_mat` qui effectue un produit matrice-matrice.
4. Implanter l'algorithme `hGCD` qui calcule la matrice de demi-PGCD de deux polynômes.
5. En déduire l'algorithme `mGCD` qui calcule la matrice de PGCD de deux polynômes.
6. En déduire l'algorithme `fGCD` de PGCD rapide.

Exercice 3. *Bonus*

1. Implanter l'algorithme `mul_karastuba` qui effectue le produit de deux polynômes à l'aide de l'algorithme de Karatsuba. *Utiliser l'algorithme dans les autres algorithmes.*
2. Implanter l'algorithme `inv_Newton` d'inversion de série formelle par itération de Newton.
3. En déduire un algorithme `div_Newton` de division euclidienne rapide de deux polynômes.