

TP ALBA séries formelles et division euclidienne

20 Octobre 2020

Pour instancier un anneau de série formelle à une variable x sur un anneau A , on utilise le code suivant :

```
B. < x >= PowerSeriesRing(A)
```

N'hésitez pas à manipuler un peu les séries formelles dans le terminal, notamment en utilisant `O` et en regardant les fonctions et les attributs de la classe `PowerSeries` avant de commencer les exercices. En particulier, que peut-on dire de leur représentation en machine ?

N'oubliez pas de tester vos fonctions en prenant différents anneaux ou corps de base. Vous pouvez générer des éléments aléatoires d'un anneau de polynôme ou de série formelle avec la méthode `A.random_element(k)` où k est le degré désiré et A l'anneau des séries formelles en question.

Exercice 1 :

Écrire la fonction `InverseMod` qui prend en entrée une série formelle et renvoie son inverse. Attention, une série formelle a un ordre de précision et son inverse doit avoir le même ordre. La fonction doit vérifier que la série formelle en entrée est inversible et renvoyer une erreur si ce n'est pas le cas. (Souvenez vous du critère d'inversibilité des séries formelles vu en cours).

On montrera d'abord que si G et H sont inversibles et que $G = H \pmod{X^N}$, alors $G^{-1} = H^{-1} \pmod{X^N}$.

Exercice 2 :

Écrire une fonction `DivEuclidNaif` et `DivEuclidRapide` qui prend en entrée deux polynômes et renvoie la division euclidienne. La première utilise l'algorithme naïf et la deuxième l'algorithme rapide. Faire une rapide comparaison sur des polynômes de degré croissant sur le corps finis de votre choix.