Université de Bordeaux
Mathématiques

Calcul formel – Master
2025–2026

FEUILLE D'EXERCICES n° 7

Travail sur machine et papier

1. Reprise du TD précédent.

Exercice 1 – [Restes chinois]

1) Résoudre dans $\mathbb{Z}$ les systèmes

$$\begin{cases} 12x - 12 \equiv 6 \bmod 33 \\ 7x + 6 \equiv 7 \bmod 13 \\ 6x - 21 \equiv 9 \bmod 54 \end{cases} \qquad \begin{cases} 15x \equiv 7 \bmod 25 \\ 8x \equiv 1 \bmod 13 \\ 7x \equiv 4 \bmod 11 \end{cases}$$

$$\begin{cases} x \equiv 5 \bmod 21 \\ x \equiv 3 \bmod 28 \\ x \equiv 1 \bmod 5 \end{cases} \qquad \begin{cases} x \equiv 3 \bmod 21 \\ x \equiv 17 \bmod 49 \\ x \equiv 1 \bmod 5 \end{cases} \qquad \begin{cases} 3x + 1 \equiv 0 \bmod 5 \\ 4x + 2 \equiv 1 \bmod 7 \\ x - 1 \equiv 1 \bmod 4 \end{cases}$$

2) Reprendre l'exercice en utilisant la commande `crt` de sage.

2. Primalité

Rappelons que si p est premier, tout a non divisible par p vérifie $a^{p-1} \equiv 1 \mod p$. Ainsi, un entier n étant donné, si l'on trouve un $a \in [[1, n-1]]$ tel que

$$a^{n-1} \not\equiv 1 \mod n, \tag{1}$$

on sait que n n'est pas premier. Ceci fournit un test de non-primalité : on prend des a au hasard entre 1 et $n-1$ et on calcule $a^{n-1} \mod n$. Dès que l'un d'entre eux vérifie (1), on sait que n est composé. Si, en revanche, au bout d'un certain nombre d'essais, (1) n'a toujours pas été vérifiée, on peut juste dire que n a des chances d'être premier. Hélas, de nombreux nombres composés peuvent passer à travers ce test, en particulier les nombres dits de Carmichael.

Définition 1. On appelle *nombre de Carmichael* tout nombre composé n vérifiant

$$a^{n-1} \equiv 1 \mod n \quad \text{pour tout } a \text{ premier à } n.$$

On admet le théorème suivant qui donne un critère pour déterminer si un nombre est de Carmichael.

Théorème 2 (Critère de Korselt)**.** *Un entier est un nombre de Carmichael si et seulement s'il est composé, sans facteur carré, et si pour tout premier p divisant n, l'entier $p-1$ divise $n-1$.*

Exercice 2 – [Exemples et Contre-Exemples]

1) Que vaut $2^{14} \mod 5$? En déduire que 2 est un témoin de non primalité de Fermat pour 15.

2) Soit a un entier premier à $561 = 3.11.17$. En utilisant la factorisation $560 = 2^4 \cdot 5 \cdot 7$, calculer a^{560} modulo chacun des entiers 3, 11 et 17 ? En déduire que 561 est un nombre de Carmichael.

3) Montrez de même que 1729 et 29341 sont des nombres de Carmichael. On pourra obtenir la factorisation à l'aide de la commande `factor` de `Sage`.

Exercice 3 – [Menteurs de Fermat]

Soit n un nombre entier composé qui n'est pas un nombre de Carmichael. On rappelle qu'un menteur de Fermat pour n est un entier $1 \leq a \leq n-1$ qui vérifie tout de même $a^{n-1} \equiv 1 \mod n$. On note M_n l'ensemble de ces menteurs de Fermat.

1) Démontrez que M_n est un sous-groupe strict de $(\mathbb{Z}/n\mathbb{Z})^\times$.

2) En déduire que son cardinal est forcément inférieur ou égal à $\varphi(n)/2$.

Exercice 4 – [Algorithme Mystère]

1) Que fait l'algorithme `Mystere` suivant ?

Algorithme 1. Mystere

Entrées: n : entier naturel non nul
Sorties: `true` ou `false`

```
F=factor(n)
if len(F)<=1 :
    return false
for f in F :
    if f[1]>1 :
        return false
for f in F :
    if (n-1) % (f[0]-1) == 0 :
        return false
return true
```

---

2) Écrire `Mystere` sur sage, et une fonction qui prend en entrée un entier N et rend en sortie la liste des N plus petits nombres de Carmichaël.

3) Dresser la liste des 30 premiers nombres de Carmichael que l'on stockera dans une liste pour la suite du travail.

Exercice 5 – [Test de Fermat]

1) Écrire une fonction en `Sage` qui prend en entrées deux entiers n et a et calcule $a^{n-1} \bmod n$. Attention à la complexité de votre implémentation !

2) Vérifiez la correction sur des petits exemples, et testez également votre algorithme sur l'entier $n = 10^{20} + 67$ et plusieurs entiers a pris au hasard.

3) À l'aide de votre fonction, programmez le test de Fermat. Votre test devra également prendre un paramètre k qui définit le nombre de tests à effectuer avant de conclure. Si au bout de k essais on n'a aucun résultat négatif, votre test doit renvoyer `True` et `False` sinon. Dans le cas où votre test renvoie `True`, on ne peut pas tout-à-fait conclure à la primalité de n (par exemple pour les nombres de Carmichael). On sait simplement que n est « probablement premier » (on dit parfois aussi *pseudopremier*).

4) Testez votre algorithme sur les nombres de Carmichael définis dans l'exercice précédent.

5) Comparez le nombre d'entiers pseudopremiers < 10000 avec le nombre d'entiers réellement premiers dans cet intervalle.

3. Arithmétique et Matrices

Exercice 6 – [Un peu de calcul matriciel]

Dans l'exercice 7, on étudie la résolution dans $\mathbb{Z}^n$ d'une équation $a_1x_1 + \cdots + a_nx_n = b$. Pour cela, nous utiliserons des matrices. C'est pourquoi cet exercice donne quelques commandes sage pour de tels calculs.

1) On peut définir un vecteur et une matrice de la manière suivante.

```
w = vector([1,1,-4])
A = matrix([[1,2,3],[3,2,1],[1,1,1]]); A
```

Remarquons d'abord que les indices commencent à 0. Si l'on tape `A[0,0]`, on obtient `1`.

2) Exécuter les commandes

```
A.det()
A*w
w*A
parent(A)
parent(w)
```

3) On peut aussi commencer par définir l'espace matriciel où se trouveront les matrices. Exécuter les commandes

```
EM=MatrixSpace(ZZ,3)
EV=VectorSpace(ZZ,3)
```

Erreur ! C'est que $\mathbb{Z}$ n'est pas un corps, on ne peut donc pas définir d'espace vectoriel sur $\mathbb{Z}$. On peut écrire à la place

```
EV=VectorSpace(QQ,3)
```

ou bien définir le module $\mathbb{Z}^3$ sur $\mathbb{Z}$.

```
EV=FreeModule(ZZ,3)
w=EV([1,1,-4])
A=EM([1,2,3,3,2,1,1,1,1])
A,w
V=EM(1)
V
V[0,1]=2
V
```

★ **Exercice 7** – [Relation de Bézout et calcul matriciel]

1) Soient deux entiers a et b tels que $(a,b) \neq (0,0)$. Soient u et v deux entiers tels que $au + bv = \mathrm{pgcd}(a,b)$. Déterminer en fonction de a, b, u, v et $d = \mathrm{pgcd}(a,b)$ une matrice $U \in \mathcal{M}_2(\mathbb{Z})$ de déterminant 1 telle que $(a,b)U = (\mathrm{pgcd}(a,b), 0)$.

2) En s'inspirant de la question précédente, montrer qu'il existe une matrice U dans $\mathcal{M}_n(\mathbb{Z})$ de déterminant 1) telle que

$$(a_1, \ldots, a_n)U = (\mathrm{pgcd}(a_1, \ldots, a_n), 0, \ldots, 0)$$

et programmer le calcul de cette matrice.

Pour cela, on peut d'abord calculer une matrice V_1 telle que

$$(a_1, \ldots, a_n)V_1 = (\mathrm{pgcd}(a_1, a_2), 0, a_3, \ldots, a_n),$$

puis une matrice V_2 telle que

$$(\mathrm{pgcd}(a_1, a_2), 0, a_3, \ldots, a_n)V_2 = (\mathrm{pgcd}(a_1, a_2, a_3), 0, 0, a_4, \ldots, a_n)$$

et itérer le processus. Alors la matrice U cherchée est égale au produit des V_i.

Application : Équations diophantiennes linéaires

On cherche les solutions entières de l'équation

$$2x_1 + 3x_2 + 5x_3 = 0. \tag{2}$$

3) Trouver une matrice U dans $\mathcal{M}_n(\mathbb{Z})$ et de déterminant 1 vérifiant $(2,3,5)U = (1,0,0)$.

4) On note $X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$. En posant $Y = UX$, expliquez comment retrouver les solutions de l'équation (2).

5) Plus généralement, si $b \in \mathbb{Z}$ et $(a_1, \ldots, a_n) \in \mathbb{Z}^n$ sont fixés, expliquer (sans le programmer) comment résoudre l'équation $\sum a_i x_i = b$ en nombres entiers (x_i) [*intercaler* $UU^{-1} = \mathrm{Id}$].

6) Résoudre les équations $1009x + 345y + 56z = 1$ et $143x + 195y + 165z = 3$.

7) Comment résoudre un système de plusieurs équations sur $\mathbb{Z}^n$? Il n'est pas demandé de programmer ce calcul.