Logiciel de calcul formel
Cours 1

Julien Lavauzelle

Université Paris 8

Licence Mathématiques

22/01/2022

Présentation

Quelques informations personnelles :

- Julien Lavauzelle, maître de conférences depuis 2020, Univ. Paris 8
- email : `julien.lavauzelle@univ-paris8.fr`
- Ma recherche : codes correcteurs, applications en cryptographie
- J'enseigne aussi :

  théorie de l'information (M1)
  cryptographie à clef publique (M1)
  algorithmes arithmétiques (M2)
  codes correcteurs (M2, à Univ. Paris 13)

Documents et ressources sur cours

La **page web** de ce cours :

`www.math.univ-paris13.fr/~lavauzelle/teaching/2021-22/calcul-formel.html`

Contiendra :

- des informations sur le cours
- les slides, des tutoriaux/aides
- les exercices de TP et certaines solutions

Le cours **Moodle** (dans la section STN/Mathématiques/L2) :

« Logiciel de Calcul Formel L2 »

→ **vous inscrire** d'ici à la semaine prochaine
le mot de passe est au tableau

La page Moodle contiendra :

- des quizz en ligne,
- les dépôts d'exercices à rendre.

Programme

12 séances de cours, le lundi de **15h30** à 18h00, salle A160

Programme provisoire du cours :

- Rappels sur `python`.
- Utilisation du notebook *Jupyter*.
- Arithmétique élémentaire.
- Polynômes.
- Matrices et systèmes linéaires.
- Calcul symbolique.
- Combinatoire.
- Algorithmes probabilistes.
- Algorithmique sur des graphes.

Évaluation

Modalités d'évaluation de l'UE :

1. **Première note : contrôle continu**
   - À chaque séance (à partir de la deuxième), une série d'exercices à faire à la maison.
   - Vous devez rendre au moins 4 séries d'exercices, 5 maximum.
     $\rightarrow$ si 5 séries rendues, je garde les 4 meilleures notes
   - Chaque série est notée sur 5 points.
   - À rendre sur Moodle.

2. **Seconde note : mini-projet**
   - Des sujets de projet vous seront proposés vers le milieu du semestre.
   - Vous devrez rendre un court rapport + les programmes que vous aurez écrit.

Note finale : 50 % note 1 + 50 % note 2

Aujourd'hui

1. Qu'est ce que le calcul formel ?

2. Sagemath

3. Rappels de python

4. Partie pratique

Plan

Définitions des termes

Qu'est ce que le **calcul formel** ?

- **But :** obtenir les **solutions d'un problème mathématique** à par des moyens informatiques
- algorithmes fondés sur des opérations mathématiques **exactes**, sur des objets à **description finie** mais pouvant être « abstraits » (symboles, variables, etc.)
- aussi appelé **calcul symbolique**, ou **calcul algébrique**.

On l'oppose au **calcul scientifique**, ou **calcul numérique** (solutions approchées, avec des nombres « flottants »).

Exemples

Quelques problèmes que l'on pourra voir en cours :

- Trouver tous les couples de rationnels $(x, y) \in \mathbb{Q}^2$ tels que $\begin{cases} 3x + y = 7 \\ x - \frac{1}{2}y = 9 \end{cases}$
- Calculer le pgcd de 123456 et 345678.
- Trouver les solutions d'une équation polynomiale de degré 4.
- Calculer la primitive d'une fonction.
- etc.

Applications :

- Aide à la preuve mathématique, « démonstration automatique »
- Cryptographie (factorisation, logarithme discret)
- Résolution d'équations différentielles

Plan

Le logiciel Sagemath

Logiciel de calcul formel pour ce cours : Sagemath

- depuis 2005 (dernière version : sagemath 9.4, août 2021)
- `https://www.sagemath.org/`

Sagemath est un logiciel **libre**, sous licence GPL :

- le code source est disponible en ligne
- on peut l'utiliser, le modifier, proposer de nouvelles versions, etc.
- plusieurs centaines de développeurs réguliers à travers le monde

« alternative à Magma, Maple, Mathematica et Matlab »

Ce que contient Sagemath

Sagemath s'appuie sur de nombreux logiciels et bibliothèques préexistantes :

- `Maxima` et `Singular` (génériques)
- `PARI/GP` et `NTL` (théorie des nombres)
- `LinBox` (algèbre linéaire)

Il s'interface également avec d'autres logiciels comme `GAP`, `Magma`, `Mathematica`, `Maple`.

Sagemath utilise `python` comme langage de base
$\rightarrow$ la syntaxe est celle de `python`

Sondage : qui a déjà utilisé `python` ?

Mode d'utilisation 1 : en ligne de commande

En ligne de commande :

1. Ouvrir un terminal : l'icône ressemble à $\longrightarrow$ 
2. Taper : `python`
3. Vous devriez obtenir quelque chose comme :

```
Python 3.9.2 (default, Apr 30 2021, 15:28:52)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

C'est le mode d'utilisation « fondamental » :

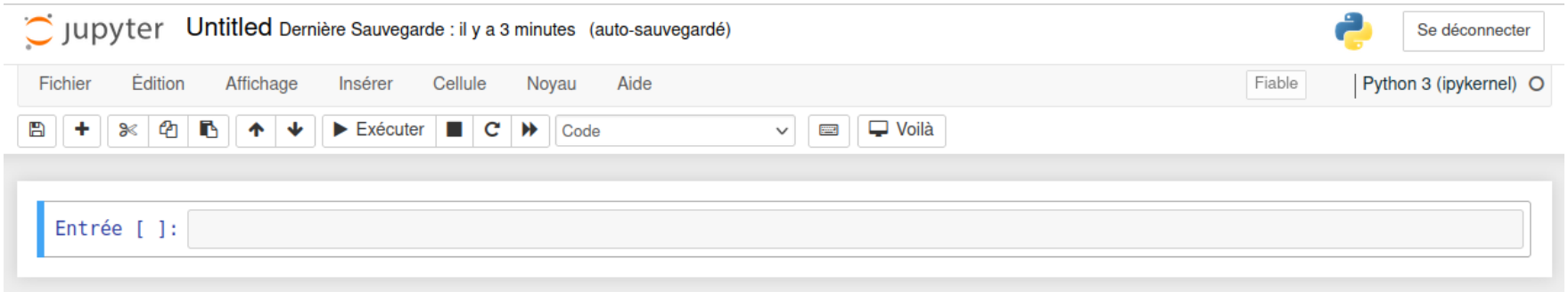
- on peut taper des instructions directement dans l'invite de commande (après les triples chevrons >>>)
- on peut éditer un fichier externe, puis l'exécuter.

Démonstration...

Mode d'utilisation 2 : par le *notebook*

C'est un mode d'utilisation avec une **interface plus confortable** visuellement :

- on édite des blocs de code dans des **cellules**,
- on peut exécuter les cellules une à une,
- l'affichage se fait directement sous la cellule.



Démonstration...

Plan

Références

Une introduction à Python 3, par Bob Cordeau et Laurent Pointal, disponible ici :

`https://perso.limsi.fr/pointal/python:courspython3`

Apprendre à programmer avec Python, par Gérard Swinnen, disponible ici :

`https://www.inforef.be/swi/python.htm`

Programmation en Python pour les sciences de la vie, par Patrick Fuchs et Pierre Poulain, disponible :

`https://python.sdv.univ-paris-diderot.fr/`

Démonstration

Passage en revue de la syntaxe de `python` et de ses fonctionnalités élémentaires

$\rightarrow$ démonstration

Plan

Partie pratique

`www.math.univ-paris13.fr/~lavauzelle/teaching/2021-22/calcul-formel.html`

Aussi accessible sur la page Moodle du cours.

- le sujet est sur la page web : **[sujet html]**
- le fichier à remplir est dans un archive `.zip` nommée **[source]**
  1. télécharger l'archive puis la décompresser où vous le souhaitez
  2. ouvrir un terminal
  3. ouvrir le notebook avec la commande :
     ```
     python -m notebook
     ```
  4. charger le fichier `tp1-sujet.ipynb` de l'archive, puis

À vous de jouer!