

# Calcul et programmation avec SageMath/Python et meilleures pratiques

Cours de l'EDMI  
25 février, 4, 11, 18 mars 2021

Sébastien Labbé  
CNRS, LaBRI, Université de Bordeaux

# Titre du cours

*Calcul et programmation avec SageMath/Python et meilleures pratiques*

Le titre se décline comme ceci :

- **Calcul et programmation** avec Python  
*Structures de données de base, les bases de la programmation, les différentes interfaces*
- Apprentissage des **meilleures pratiques**  
*Wilson et al., Best Practices for Scientific Computing, PLoS Biology 12 (2014) 1, <https://doi.org/10.1371/journal.pbio.1001745>*
- Basé sur une expérience acquise via le développement de **SageMath**  
*logiciel de mathématiques développé depuis 2006*

Tout ça en 4 cours de 3h.

<http://www.slabbe.org/Enseignements/2021-02-cours-edmi-python/>

# Les 4 séances

Séance 1 (25 février 2021) : Python

- Calculatrice Python et structures de données de base
- Notebook Jupyter et programmation en Python

Séance 2 (4 mars 2021) : GPL, git, class

- JupyterLab et les classes en Python
- La GPL, le terminal (bash), logiciel de contrôle de version (git), serveur git, premiers commits dans un projet, création d'une clé SSH.

Séance 3 (11 mars 2021) et Séance 4 (18 mars 2021) :

- Structures de données de base : `tuple`, `dict`, `set`, `frozenset`
- Éditeur de texte, logiciel de création de fichiers (make)
- Documentation et tests (doctest), compilation de la documentation (ReStructuredText, sphinx), création d'un paquet Python et reproductibilité.
- Bibliothèques Python (Cython, Pandas, Matplotlib, SageMath, argparse, ...) selon les goûts.

Si c'est trop ambitieux, on adaptera le rythme.

# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.

# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.



# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.



# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.



# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.





# Logiciels libres

Les licences de logiciels libres, tel que la **licence publique générale** de GNU, garantissent quatre types de liberté :

- La liberté d'**utiliser** le programme.
- La liberté de **lire le code source**.
- La liberté d'**améliorer le logiciel**.
- La liberté de **redistribuer le programme**.



La gestion de ces libertés s'effectue via un logiciel de contrôle de versions.

Linus Torvalds crée Git en 2005 pour le développement du noyau Linux.

<https://en.wikipedia.org/wiki/Git>

# Préalables pour aujourd'hui

- Vous avez installé JupyterLab et vous pouvez l'ouvrir  
<http://www.slabbe.org/blogue/2021/02/installation-de-python-jupyter-et-jupyterlab>
- Vous avez installé git et vous pouvez l'ouvrir "git --version" fonctionne dans un terminal dans JupyterLab
- Vous avez activé votre compte sur <https://gitub.u-bordeaux.fr/> ou sinon sur [gitlab.com](https://gitlab.com)

# Aujourd'hui – 4 mars 2021 – git et classes Python

La liberté d'**utiliser** le programme.

- Par exemple : `https://live.sympy.org/`

La liberté de **lire le code source**.

- JupyterLab
- `git clone`, `git shortlog`, `git log`, `git show`, `git lg`

La liberté d'**améliorer le logiciel**.

- Petit cours sur les classes en Python (les fruits !)

```
class Fraise
```

- Modifier le code source localement

```
git status, git diff, git add, git commit
```

La liberté de **redistribuer le programme** :

- Cloner un dépôt sur son compte (le dépôt "Matrice")

```
https://gitub.u-bordeaux.fr/
```

- Modifier et partager le code source modifié en groupe

```
git push, git pull
```

# Références

Sur git :

- <https://thor.enseirb-matmeca.fr/ruby/docs/repository/git>  
en français, avec vidéos explicatifs  
par David Renault et d'autres professeurs de l'ENSEIRB de Bordeaux
- <https://github.com/rougier/URFIST-git-course>  
en anglais  
par Nicolas Rougier (cours donné en mars 2018)
- <https://www.linuxfoundation.org/blog/10-years-of-git-an-interview-with-git-creator-linus-torvalds/>

Sur la programmation orientée objet en Python :

- <https://docs.python.org/fr/3/tutorial/classes.html>
- <https://gitlab.com/seblabbe/slabbe/-/blob/develop/slabbe/fruit.py>  
module simple Python pour illustrer l'héritage des classes en Python
- <https://courspython.com/classes-et-objets.html>