

TP1__2025

December 11, 2024

0.1 L1 informatique – Calcul scientifique

0.2 TP 1: prise en main du jupyter notebook et premiers pas avec sympy

0.2.1 NOM :

0.2.2 Prénom :

Objectifs du TP :

- Savoir exécuter jupyter notebook, lancer/sauvegarder un notebook, importer une librairie.
- Introduction à sympy , quelques calculs et graphes de base

Ce TP est à finir (même s’il n’est pas à rendre); ce sera très utile pour les exos noté et le TP noté.

Quelques conseils de base : Pour chaque fiche de TP, nous utiliserons un calepin jupyter notebook. Il est conseillé de créer un répertoire CalculScientifique, puis un sous-répertoire pour chaque TP (e.g., TP1, TP2, TP3...). En général, une archive, que vous pouvez télécharger sur ecampus, est composée du fichier TP_X.ipynb qui contient les commandes (pratiquement) pré-remplies correspondant au TP X ainsi que les ressources nécessaires. Il faut *renommer* le fichier .ipynb sous la forme TP_X_Nom_Prenom.ipynb. Dans un répertoire TP1, vous stockerez donc : - Le fichier TP_1_Nom_Prenom.ipynb - Le fichier TP_1_Nom_Prenom au format PDF - Les fichiers de données (qui y seront téléchargés)

0.3 I) Jupyter Notebook

0.3.1 Mise en forme

Ajoutez une nouvelle cellule tout en haut du jupyter notebook sous le titre. Puis, vous ajouter votre nom et votre prénom dans cette cellule sous format **Markdown**. Votre nom doit être en gras. On pourra cliquer sur une cellule pour voir le “code” pour mettre différents niveaux de titres.

0.4 II) Utilisation de Sympy

Il s’agit de “faire faire” des maths à python grâce à sympy. Il s’agit pour cette première expérience de travailler avec de polynômes et des systèmes , comme en algèbre au semestre 1.

Quelques outils à utiliser: display, pprint, factor, expand, subs, eq, solve, diff, div et à retenir...

0.5 Exercice 1 - premiers calculs avec sympy

Ne pas oublier de commencer par importer sympy

```
[2]: import sympy as sp
```

Exemple: on définit $\sqrt{2}$ puis on l'affiche de plusieurs manières (print, sp.pprint, display)

```
[3]: e= sp.sqrt(2)
print("avec print :")
print("e=",e)

print("\navec sp.pprint :")
sp.pprint(e)

print("\navec display :")
display(e)
```

avec print :
 $e = \sqrt{2}$

avec sp.pprint :
 $\sqrt{2}$

avec display :
 $\sqrt{2}$

1. Définir le symbole x puis les expressions:

$$e_1 = x^3 - 8$$

$$e_2 = (x - 2)(x^2 - 8)$$

On essayera plusieurs affichage (print, sp.pprint et display)

```
[5]: # Définir le symbole x
# A compléter ...
x= ...

# Définir les expressions e1, e2 et e3
# A compléter ...

e1= ...
e2= ...

print("Avec print :")
print(e1)
print(e2)

print("\nAvec sp.pprint :")
```

```

sp.pprint(e1)
sp.pprint(e2)

print("\nAvec display :")
display(e1)
display(e2)

```

avec print :

```

e1= x**3 - 8
e2= (x - 2)*(x**2 - 8)

```

avec sp.pprint :

```

3
x  - 8
      2
(x - 2) x  - 8

```

avec display :

$$x^3 - 8$$

$$(x - 2)(x^2 - 8)$$

2. calculer la valeur de e_1 pour $x=2$ et pour $x=-2$.

[6]: *# A compléter ...*

```

pour x=2  e1 vaut 0
pour x=-2 e1 vaut -16

```

3. Factoriser e_1 et développer e_2 .

[7]: *# A compléter ...*

$$\sqrt{2}$$

$$x^3 - 8$$

4. Simplifiez les expressions suivantes :

$$\frac{x^2 + 2x + 1}{x + 1}$$

$$\sin^2(x) + \cos^2(x)$$

[8]: *# A compléter ...*

```

expression1 = ...

```

```

expression_simplifiee1 = ...
print("Expression simplifiée 1 :", expression_simplifiee1)

expression2 = ...

expression_simplifiee2 = sp.simplify(expression2)
print("Expression simplifiée 2 :", expression_simplifiee2)

```

Expression simplifiée 1 : $x + 1$

Expression simplifiée 2 : 1

5. Calculer et factoriser $e = e_1 + e_2$ et $ee = e_1 - e_2$

[10]: # A compléter ...

pour e

$$x^3 + (x - 2)(x^2 - 8) - 8$$

et en factorisant

$$2(x - 2)(x - 1)(x + 2)$$

pour ee

$$x^3 - (x - 2)(x^2 - 8) - 8$$

et en développant

[10]: $2(x - 2)(x + 6)$

6. Déterminer les racines de e et ee

[30]: # A compléter ...

Racines de e : [-2, 1, 2]

[29]: # A compléter ...

Racines de ee : [-6, 2]

7. Créer les équations $e=0$ et $ee=0$

[26]: # A compléter ...

```

eq1= ...
eq2= ...

print("Equation 1 :")
display(eq1)
print("Equation 2 :")
display(eq2)

```

Equation 1 :

$$x^3 + (x - 2)(x^2 - 8) - 8 = 0$$

Equation 2 :

$$x^3 - (x - 2)(x^2 - 8) - 8 = 0$$

8. Résoudre le système

$$\begin{cases} e = 0 \\ ee = 0 \end{cases}$$

```
[33]: print ("Solution :")
      # A compléter ...
```

Solution :

```
[33]: [{x: 2}]
```

0.6 Exercice 2 - résolution de systèmes avec sympy

A) Résoudre avec sympy les systèmes suivants :

$$1) \begin{cases} 5x + 2y + 3z = -2 \\ 2x - 2y + 5z = 0 \\ 3x + 4y + 2z = 10 \end{cases}$$

$$2) \begin{cases} 2x - y + 3z = 9 \\ 3x - 5y + z = -4 \\ 4x - 7y + z = 5 \end{cases}$$

$$3) \begin{cases} x + y - z = 2 \\ 2x + 3y + 2z = 1 \\ x + 2y + 3z = -1 \end{cases}$$

```
[ ]: # Définir les variables symboliques
      # A compléter ...
```

```
# Définir les systèmes d'équations
# A compléter ...
```

```
[34]: # A compléter ...

      # Afficher les solutions
      print("Solution du système 1:")
```

```
print(solution1)
```

Solution du système 1:
{x: -26/7, y: 53/14, z: 3}

```
[15]: # A compléter ...

print("\nSolution du système 2:")
print(solution2)
```

Solution du système 2:
[]

```
[16]: # A compléter ...

print("solution du systeme 2 ")
sp.solve([eq1,eq2,eq3])
```

solution du systeme 2

```
[16]: []
```

```
[17]: # A compléter ...

print("\nSolution du système 3:")
print(solution3)
```

Solution du système 3:
{x: 5*z + 5, y: -4*z - 3}

B) On considère le système:

$$\begin{cases} x + my = -3 \\ mx + 4y = 6 \end{cases}$$

où m est un paramètre.

Résoudre ce système avec sympy et remarquer que la solution donnée est la solution la plus générale mais qu'il n'y a pas de discussion selon les valeurs de m.

Remarque: attention à bien avoir les solutions pour x et y en fonction de m .

```
[19]: # A compléter ...
```

```
[19]: {x: 6/(m - 2), y: -3/(m - 2)}
```

0.7 Exercice 3 - polynômes avec sympy

On considère le polynome $Q(x) = x^5 - 4x^4 + 4x^3 + 2x^2 - 5x + 2$

1) Définir et afficher q.

```
[36]: # A compléter ...
```

$$x^5 - 4x^4 + 4x^3 + 2x^2 - 5x + 2$$

2) Calculer les dérivées première, seconde et troisième et quatrième de q pour x=1.

On en déduit que $(x-1)^3$ divise Q. Calculer le quotient (avec sp.div)

```
[38]: # A compléter ...

print("Les 4 valeurs : ", r1,r2,r3,r4)

# A compléter ...
quotient=...
print("Le quotient :")
display(quotient)
```

Les 4 valeurs : 0 0 0 -12

Le quotient :

$$x^2 - x - 2$$

3) On remarque que 2 est une racine. Faire la division du quotient par x-2

```
[22]: # A compléter ...
quotient2=...
display(quotient2)
```

$$x + 1$$

4) On a donc factorisé Q. Afficher le résultat.

Retrouver ce résultat avec factor

```
[23]: # A compléter ...
```

factorisation obtenue avec divisions

$$(x-2)(x-1)^3(x+1)$$

factorisation avec sp.factor

```
[23]: (x-2)(x-1)^3(x+1)
```

Remarque: sympy ne peut pas factoriser n'importe quel polynôme: Par exemple pour $R(X) = 4X^4 + 4X^3 - 29X + 21$, on n'obtient pas une factorisation complète (alors qu'en théorie c'est possible).

[6] :

[6] : $(x - 1)(4x^3 + 8x^2 - 21x - 21)$