



Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

# Introduction aux bibliothèques numpy et matplotlib



## Objectif :

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Donner une représentation graphique pour répondre à un problème numérique.

Introduction  
à la  
bibliothèque  
matplotlib

Les commandes permettant l'affichage d'un graphique se trouve dans la bibliothèque `matplotlib.pyplot`

Introduction  
à la  
bibliothèque  
numpy

Pour réaliser une représentation graphique, il sera souvent utile/nécessaire de manipuler des variables de type `array` disponibles dans la bibliothèque `numpy`.

Les outils présents dans ces bibliothèques peuvent servir dans d'autres matières et en TIPE.

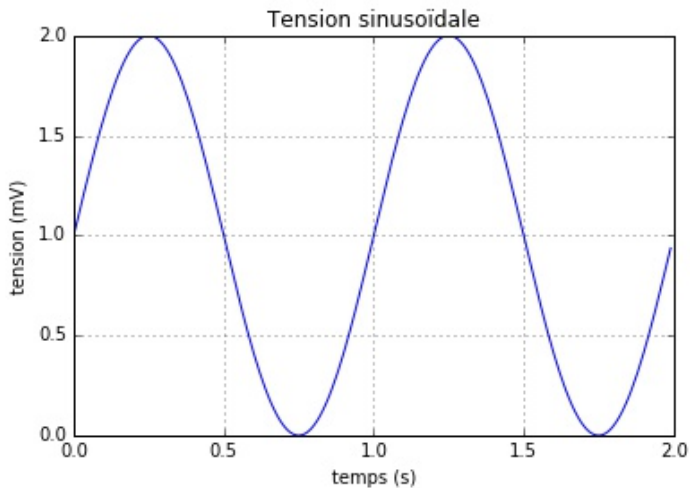


## Exemples :

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy



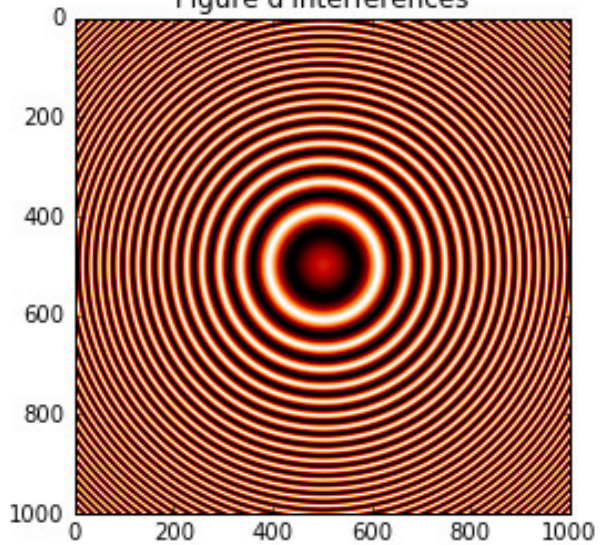


Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

Figure d'interférences

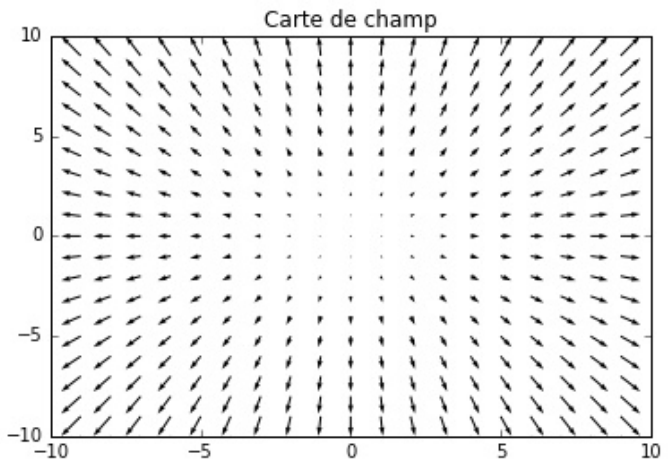




Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy



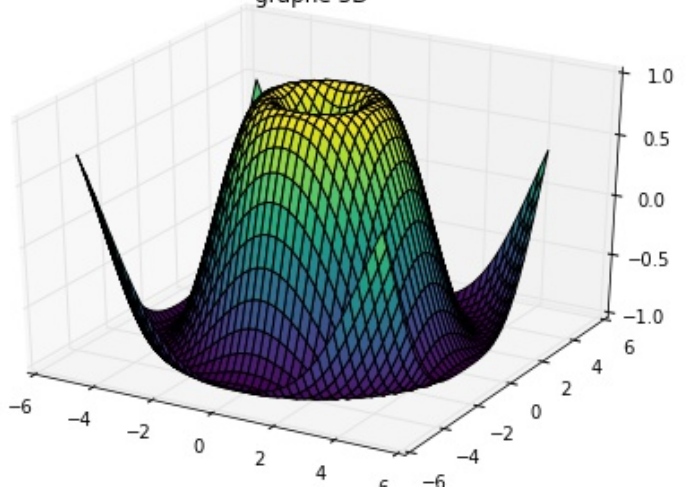


Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

graphe 3D





Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

1

Introduction à la bibliothèque matplotlib



## Premier graphe

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

Il n'est généralement pas nécessaire d'importer toute la bibliothèque. Il est très courant que, parmi les différentes syntaxes possible, il soit choisi :

```
import matplotlib.pyplot as plt
```

```
x = [1, 2, 3, 4]  
y = [-1, 0, 2, 2]  
plt.plot(x, y)  
plt.show()
```



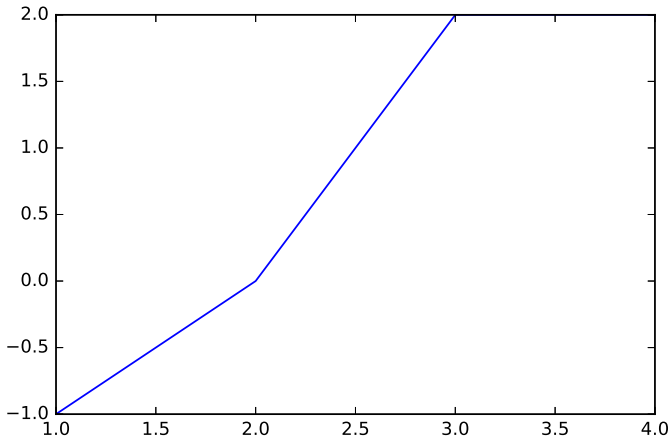


## Premier graphe.... pas très beau

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy





## Soigner la présentation...

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

```
import matplotlib.pyplot as plt
```

```
x = [1, 2, 3, 4]
```

```
y = [-1, 0, 2, 2]
```

```
plt.plot(x, y, 'ob', label="des points!")
```

```
plt.grid()
```

```
plt.xlabel('abscisses')
```

```
plt.ylabel('ordonnees')
```

```
plt.legend(loc='upper left')
```

```
plt.xlim(0, 5)
```

```
plt.ylim(-2, 3)
```

```
plt.title('Mon premier graphe')
```

```
plt.show()
```

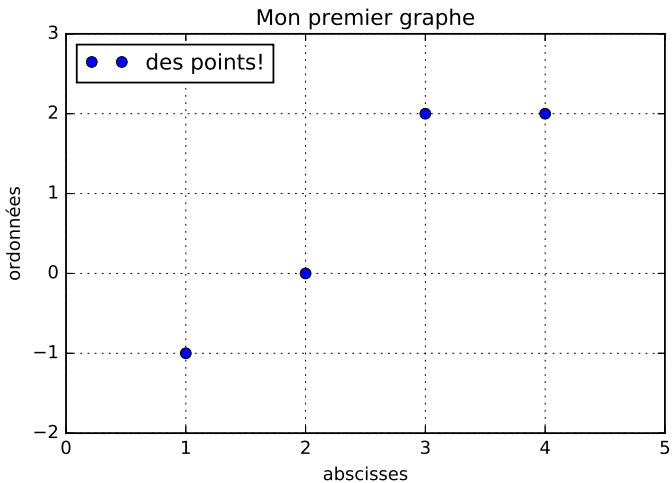


...pour obtenir un graphe bien habillé

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy





## Et ensuite ?

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy



Comment tracer le graphe représentant la fonction  $y = \sin(x)$  ?

- ▶  $\sin(x)$  n'est pas connu dans les bibliothèque de base de Python.
- ▶ comment faire quand on veut « beaucoup de points » ?



Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

2

Introduction à la bibliothèque numpy



## Que trouve-t-on dans numpy ?

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

La bibliothèque numpy est généralement importée avec la syntaxe :

```
import numpy as np
```

Cette bibliothèque permet de manipuler des variables de type `array`.

Un `array` est un tableau de données numériques. Ce type de variable possède des propriétés très pratiques pour l'analyse numérique.

```
liste = [1, 2, 3]  
tableau = np.array([1, 2, 3])
```

```
print(type(liste))  
print(type(tableau))
```



## Quelques propriétés

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

```
print (2* liste )  
print (2* tableau )
```

```
print (2+ liste )  
print (2+ tableau )
```

```
print (2/ liste )  
print (2/ tableau )
```

```
print ( liste+liste )  
print ( tableau+tableau )
```

```
print ( liste*liste )  
print ( tableau*tableau )
```



## Quelques fonctionnalités pratiques

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

```
print(np.linspace(2,10,1000))
```

Introduction  
à la  
bibliothèque  
matplotlib

```
print(np.arange(-1.2,2,0.1))
```

Introduction  
à la  
bibliothèque  
numpy

```
print(np.zeros(3))
```

```
print(np.zeros((2,4)))
```

```
print(np.sin(np.pi/2))
```





## Utilisation combinée de matplotlib.pyplot et numpy

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy



Proposer une série de commandes permettant de tracer le graphe représentant la tension sinusoïdale présentée en exemple 1 ?



## Pour aller plus loin et s'adapter à ses besoins

Introduction  
aux  
bibliothèques  
numpy et  
matplotlib

Introduction  
à la  
bibliothèque  
matplotlib

Introduction  
à la  
bibliothèque  
numpy

- ▶ Penser à consulter les aides comme `help(numpy)` et trouver des fonctionnalités intéressantes pour votre problème.
- ▶ Consulter des tutoriels et des exemples en ligne :
  - ▶ <https://matplotlib.org/>
  - ▶ <https://www.labri.fr/perso/nrougier/teaching/matplotlib/>