

Chapitre 2

Logiciels de post traitement

GNUPLOT

But

Le but de ce chapitre est de vous fournir une connaissance de base des principes utilisés pour tracer de simples graphiques 2D à l'aide du logiciel *GNU PLOT*.

À la fin de cette partie du chapitre 2, vous serez en mesure de tracer des graphiques et les sauvegarder dans des fichiers image à partir de fichiers de résultats générées par d'autres programmes.

GNU PLOT

- *GNU PLOT* est un logiciel de tracé gratuit et open source qui est largement utilisé dans les applications scientifiques depuis 1986.
- *GNU PLOT* permet de tracer rapidement des courbes à partir d'une équation simplement en lignes de commande.
- *GNU PLOT* est un puissant outil pour visualiser des données scientifiques provenant d'un fichier de données de format texte.
- *GNU PLOT* peut afficher directement un graphique dans une image,
- *GNU PLOT* peut aussi enregistrer le graphique dans un fichier image en divers formats.

Installation

- La version la plus récente de **GNU PLOT** est la version 5.4.1 (décembre 2020).
- Elle est téléchargeable depuis le lien suivant :
< <https://sourceforge.net/projects/gnuplot/files/gnuplot/5.4.1/gp541-win64-mingw.exe/download> >
- Exécutez le fichier téléchargé et laissez-le s'exécuter en tant qu'administrateur.
- L'emplacement d'installation par défaut de **GNU PLOT** sous Windows est : **C:\Program Files\gnuplot**

Lancer GNU PLOT

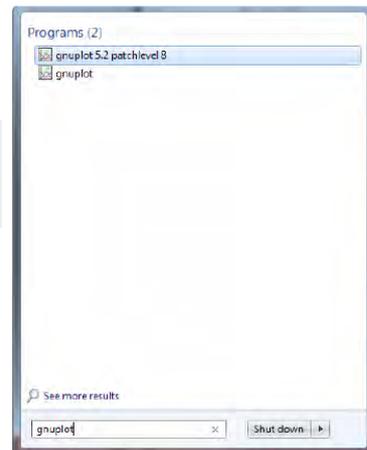
- Un **double-clic** sur l'icône de raccourci déposée sur le Bureau pour lancer l'application **GNU PLOT**.



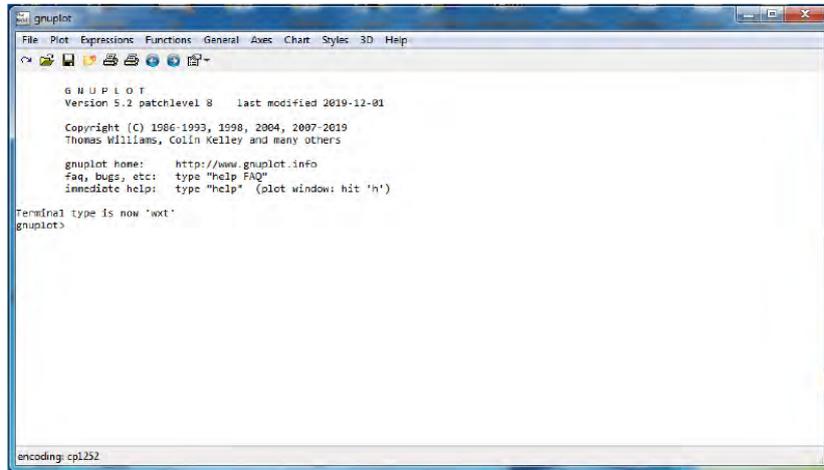
- Ouvrez le menu de démarrage de Windows.
- Saisissez **gnuplot** dans la barre de recherches.
- sélectionner

 gnuplot 5.2 patchlevel 8

- **GNU PLOT** peut être lancé depuis les fichiers **gnuplot.exe** et **wgnuplot.exe**, qui se trouvent dans le répertoire **bin** de la distribution **GNU PLOT**
< **C:\Program Files\gnuplot\bin** >.



Lancer GNUPLOT



- Lorsque **GNUPLLOT** est ouvert, on a accès à l'invite de commande (prompt) `gnuplot>` dans lequel on peut taper **un** ou **plusieurs commandes** qui seront interprétées et exécutées par **GNUPLLOT**.

Lancer GNUPLOT

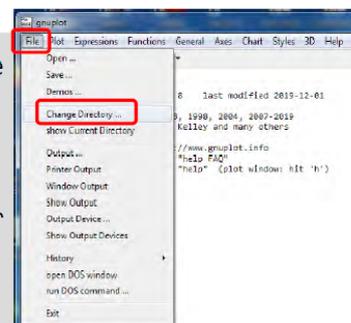
Procédures à suivre avant de travailler avec GNUPLLOT

- Il faut créer un dossier (repertoire de travail) afin de stocker et de classer vos fichiers du TP .
- Nommez Le répertoire de travail en : `TP_gnuplot` 
- Il est préférable de créer ce répertoire sur le Bureau.

Une fois **GNUPLLOT** est lancé, il faut **toujours** se positionner dans le répertoire de travail.

File > Change Directory >

Utilisez ensuite l'explorateur Windows, et naviguer jusqu'au dossier `TP_gnuplot` .



Introduction

- **GNUPLLOT** exécute des instructions une à une depuis l'invite de commande `gnuplot>` .
- **GNUPLLOT** exécute aussi des instructions une à une contenues dans un ou plusieurs fichiers **scripts**.
- Ces instructions permettent de produire des représentations **graphiques** en **deux** ou **trois dimensions** à partir de **fonctions** numériques ou de **fichiers données**.

Introduction

Dans **GNUPLLOT** :

- Les chaînes de caractères doivent être indiquées entre **quottes** (simple ou double).

Exemples

```
gnuplot> load "filename"  
gnuplot> cd 'dir'
```

- Une suite de commandes peut être étendue sur plusieurs lignes en terminant chaque ligne (sauf la dernière) par un **antislash** (\).
- La commande **help** donne accès à l'aide de **GNUPLLOT**.
- Les commandes **exit** et **quit** permettent de fermer **GNUPLLOT**.
- La commande **clear** permet d'effacer l'écran.
- les commandes passées peuvent être récupérées de l'historique et réutilisées grâce aux flèches de déplacements.

commandes principale

- La commande principale pour tracer des graphiques **2D** avec **GNUPLLOT** est :
`gnuplot> plot` est
- La commande principale pour tracer des graphiques **3D** avec **GNUPLLOT** est :
`gnuplot> splot`
- Les commande `plot` et `splot` peuvent être utilisées soit dans l'invite de commande ou bien dans un fichier script.
- On peut accéder à la documentation de l'aide de **GNUPLLOT** avec l'instruction suivante:
`gnuplot> help plot`

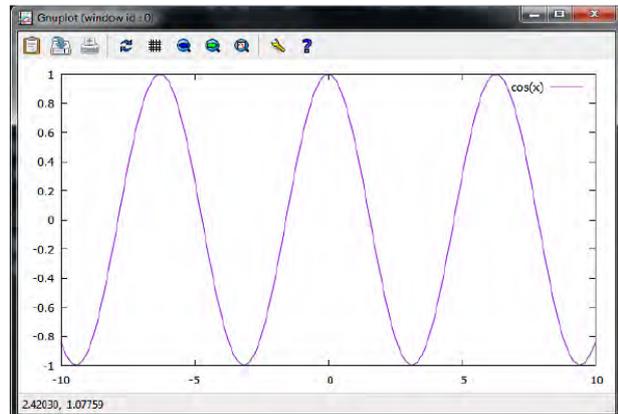
`plot` - à partir d'une fonction

- Par exemple, pour tracer les graphiques des fonctions à l'aide de **GNUPLLOT**, il suffit d'utiliser la commande `plot` avec la fonction à tracer.

Exemple

```
gnuplot> plot cos(x)
```

Cette commande devrait faire apparaître une fenêtre dans laquelle on peut voir le tracé de la fonction *cosinus*.

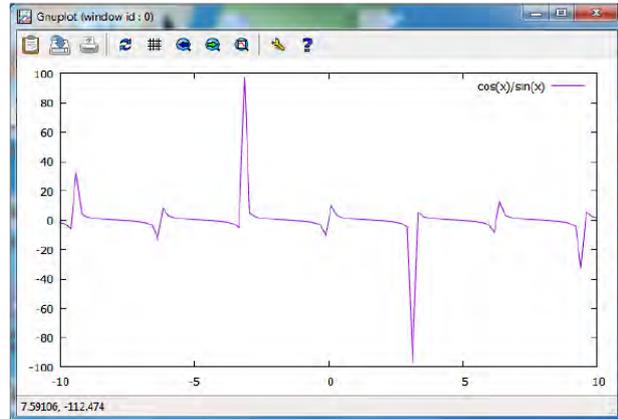


plot - à partir d'une fonction

- Pour pouvoir tracer toutes les fonctions, il est possible de combiner les fonctions pré-définies dans **GNUPLOT** en utilisant la syntaxe habituelle en mathématiques.

Exemple

```
gnuplot> plot cos(x)/sin(x)
```

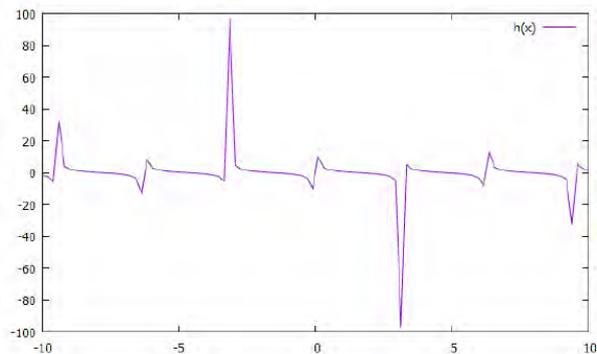


plot - à partir d'une fonction

- Pour bien gérer le tracé des fonctions, il est préférable de les définir dans un premier temps.

Exemple

```
gnuplot> f(x) = cos(x)  
gnuplot> g(x) = sin(x)  
gnuplot> h(x) = f(x)/g(x)  
gnuplot> plot h(x)
```



plot - à partir d'une fonction

- **GNU PLOT** est flexible, il permet d'utiliser les mêmes expressions mathématiques valides en langages *C*, *FORTRAN*, *Pascal*, ou *BASIC*.

Liste de quelques opérateurs binaires et leurs significations

Symbole	Exemple	Explication
**	a**b	puissance
*	a*b	multiplication
/	a/b	division
+	a+b	addition

- Pour plus d'aide sur les opérateurs binaires tapez :

`gnuplot> help operators Binary`

plot - à partir d'une fonction

Liste de quelques fonctions et leurs significations

Code	Fonction mathématique
<i>abs(x)</i>	valeur absolue d'un nombre réel x
<i>acos(x)</i>	arc cosinus
<i>asinh(x)</i>	arc sinus hyperbolique
<i>besy0(x)</i>	fonction de Bessel y0
<i>cosh(x)</i>	cosinus hyperbolique
<i>log(x)</i>	logarithme népérien
<i>sinh(x)</i>	sinus hyperbolique
<i>tan(x)</i>	tangente
<i>acosh(x)</i>	arc cosinus hyperbolique

Code	Fonction mathématique
<i>atan(x)</i>	arc tangente de x
<i>atanh(x)</i>	arc tangente hyperbolique
<i>cos(x)</i>	cosinus
<i>exp(x)</i>	exponentielle
<i>log10(x)</i>	logarithme décimal
<i>pi</i>	nombre π (3,14...)
<i>sin(x)</i>	sinus
<i>sqrt(x)</i>	racine carrée
<i>tanh(x)</i>	tangente hyperbolique

- Pour plus d'aide sur les fonctions, tapez :

`gnuplot> help expressions functions`

plot - à partir d'une fonction

- Il est possible de tracer plusieurs fonctions sur le même graphique.
- Dans ce cas, il suffit d'écrire la liste des fonctions séparée par une **virgule**, avec une seule commande **plot**.

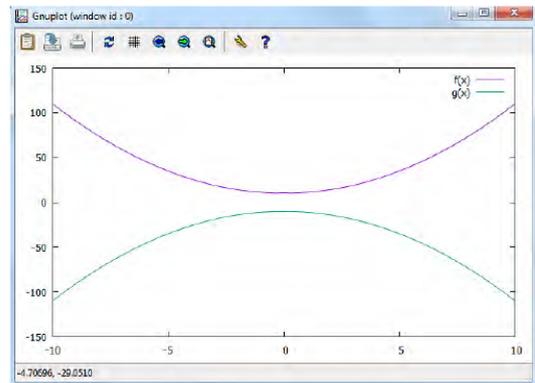
Exemple

Tracez les courbes d'équations suivantes :

$$f(x) = x^2 + 10$$

$$g(x) = -x^2 - 10$$

```
gnuplot> f(x)=x**2 + 10  
gnuplot> g(x) = -x**2 -10  
gnuplot> Plot f(x) , g(x)
```



plot - à partir des Fichiers de données

- **GNUPLLOT** permet de tracer des courbes à partir d'un ou **plusieurs fichiers de données**.
- Ces fichiers de données sont au **format texte**, contenant un ensemble de points disposés en **colonnes**, et séparés par un ou plusieurs espaces.
- Toute ligne du fichier commençant par le caractère # est ignorée par **GNUPLLOT**, c'est une ligne dite de commentaire.
- les nombres peuvent être écrits sans point décimal (pour les nombres entiers : 141 , -1412), avec le point décimal (pour le type réel : -141.4, 142.5), ou en notation à point flottant (par exemple -141.42e-02).

plot - à partir des Fichiers de données

➤ **GNU PLOT** Trace automatiquement un fichier de données simple par la commande

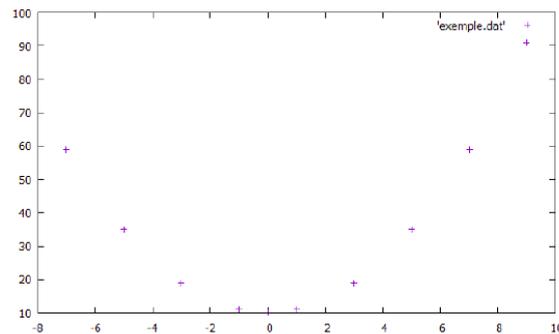
```
gnuplot> plot 'nom_du_fichier'
```

Exemple

Soit le fichier de données suivant (*exemple_1.dat*) à créer et sauvegarder dans votre répertoire de travail à l'aide de l'éditeur de texte **Notepad++**

< <https://notepad-plus-plus.org/downloads/> >.

```
# fichier = exemple_1.dat
#c1      c2
-7       59
-5       35
-3       19
-1       11
0        10
1        11
3        19
5        35
7        59
9        91
```



plot - à partir des Fichiers de données

➤ Par la commande :

```
gnuplot> plot 'exemple_1.dat'
```

- **GNU PLOT** va produire un graphique en utilisant les paramètres par défaut.
- Par défaut, **GNU PLOT** va produire un tracé par points en utilisant les deux colonnes du fichier de données '*exemple_1.dat*'.
- Le tracé est fait à partir des données de la 2^{ème} colonne (x^2+10) en fonction de la 1^{ère} colonne (x).
- Dans le graphique, les intervalles sont choisies en fonction des valeurs extrêmes des données.

plot – Styles de traçage

- **GNUPLLOT** offre plusieurs types de styles de traçage des courbes des fonctions mathématiques et de données.

Voir : `gnuplot> help plotting styles`

- Ces styles ont pour but d'affiner les graphiques et de les rendre plus clairs et plus captivants possible.

- La commande `with` permet d'utiliser le style de traçage choisi .

Voir : `gnuplot> help with`

Exemple

```
gnuplot> plot cos(x) with points
```

```
gnuplot> plot cos(x) with lines
```

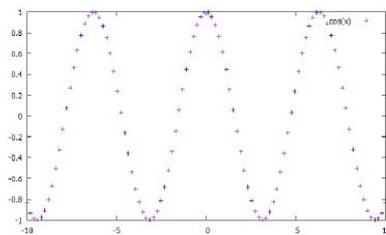
```
gnuplot> plot cos(x) with boxes
```

```
gnuplot> plot cos(x) with impulses
```

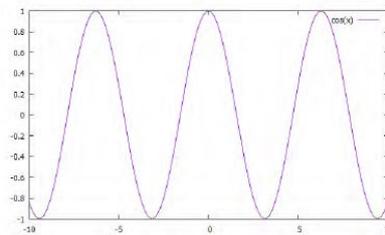
```
gnuplot> plot cos(x) with steps
```

```
gnuplot> plot cos(x) with linespoints
```

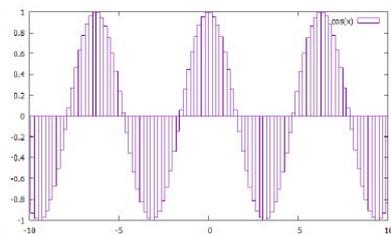
plot – Styles de traçage



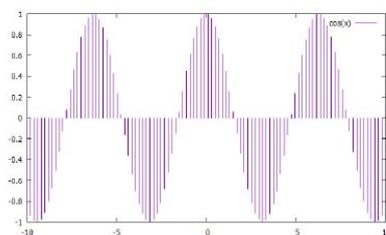
`gnuplot> plot cos(x) with points`



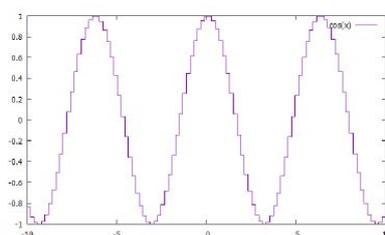
`gnuplot> plot cos(x) with lines`



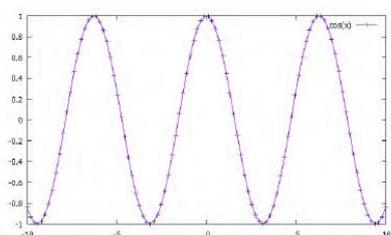
`gnuplot> plot cos(x) with boxes`



`gnuplot> plot cos(x) with impulses`



`gnuplot> plot cos(x) with steps`



`gnuplot> plot cos(x) with linespoints`

plot – Styles de traçage

- Si le style n'est pas défini (la commande *with* n'est pas utilisée) , le traçage se fait avec le style par défaut pré-défini par **GNUPLOT** :
 - le traçage en style lignes (*Lines*) est utilisé pour les fonctions, et
 - le traçage en style points (*points*) pour les données lues à partir d'un fichier.
- les valeurs par défaut pour tracer les fonctions et les données peuvent être modifiées et définies globalement par la commande *set style* .

Exemple

```
gnuplot> set style function linespoints
```

```
gnuplot> set style function points
```

```
gnuplot> set style function impulses
```

```
gnuplot> set style data lines
```

```
gnuplot> set style data linespoints
```

```
gnuplot> set style data impulses
```

plot – Styles de traçage

- En plus, **GNUPLOT** propose de nombreuses options pour **affiner** les graphiques.
- Ces options permettent de changer les paramètres du graphique (style du graphique), tel que :

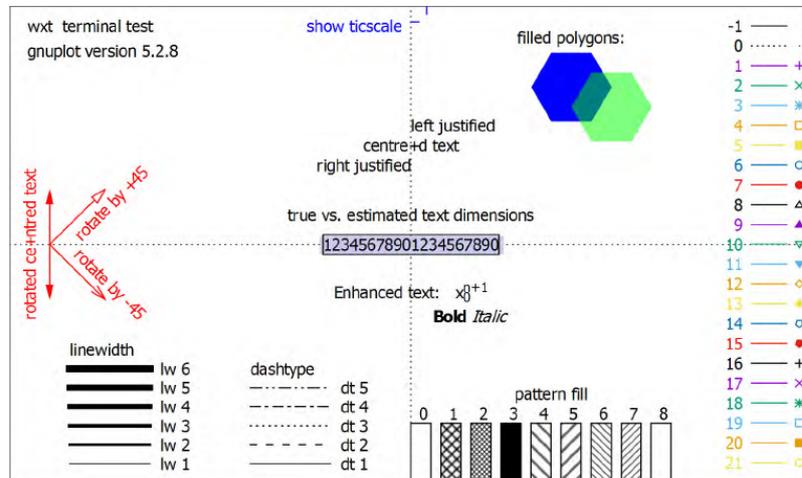
- Type de la ligne
- Largeur de la ligne
- Couleur de la ligne
- Type de points (symbole)
- Taille de points (symbole)

<i>linecolor</i>	<i>lc</i>
<i>linespoints</i>	<i>lp</i>
<i>linestyle</i>	<i>ls</i>
<i>linetype</i>	<i>lt</i>
<i>linewidth</i>	<i>lw</i>
<i>pointinterval</i>	<i>pi</i>
<i>pointsize</i>	<i>ps</i>
<i>pointtype</i>	<i>pt</i>
<i>textcolor</i>	<i>tc</i>

Abréviations supportées par les options de gnuplot.

plot – Styles de traçage

- Liste des valeurs possibles pour les attributs `linetype`, `pointtype`, `linecolor`, ..., et autres.



- Ce graphique est obtenu par la commande : `gnuplot> test`

plot – Styles de traçage

- De plus, **GNUPLLOT** offre la possibilité de définir un ensemble d'options dans une seule commande `linestyle`.
- Chaque commande `linestyle` est définie par un indice (un nombre entier) qui lui est attribué au moment de sa création.
- La commande pour créer un nouveau style de ligne est `set style Line indice Options`, ou simplement par `set ls indice Options`.

Exemple

```
gnuplot> set style Line 1 Linetype 5 Linewidth 3 pointtype 7 pointsize 2 Linecolor 6
```

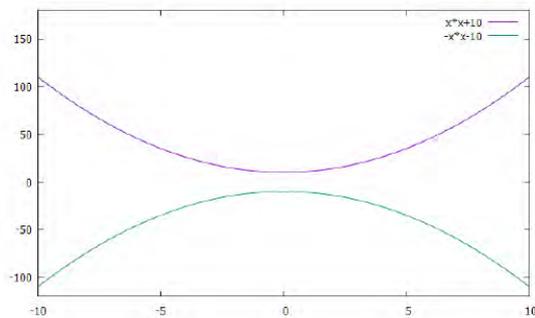
```
gnuplot> set ls 1 lt 5 lw 3 pt 7 ps 2 lc 6
```

- la commande `test` permet d'afficher les styles et éléments utiles disponibles pour le terminal courant.

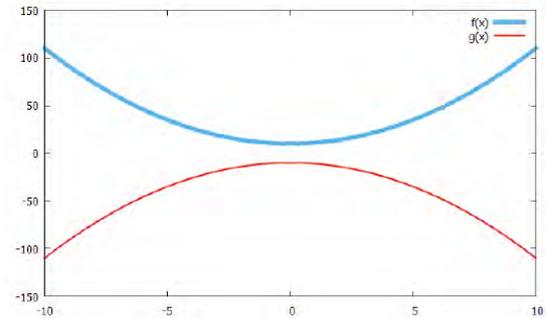
```
gnuplot> test
```

plot - Styles de traçage

Exemple



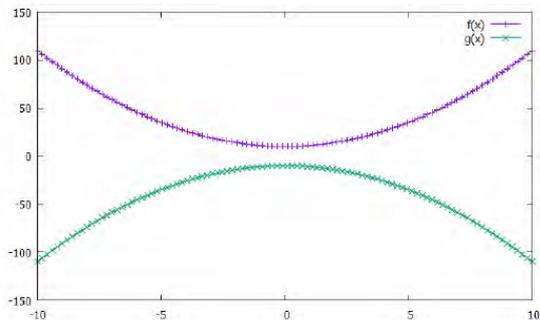
```
gnuplot> f(x)=x**2 + 10
gnuplot> g(x) = -x**2 -10
gnuplot> plot f(x) , g(x)
```



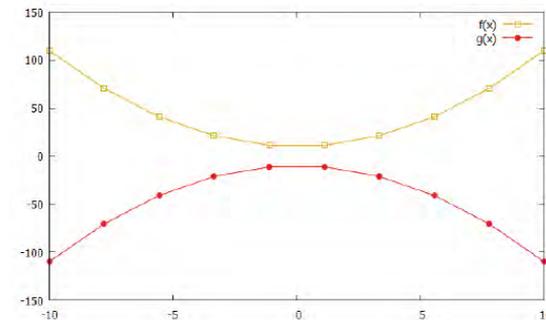
```
gnuplot> set ls 1 lt 3 lw 5
gnuplot> set ls 2 lt 7 lw 2
gnuplot> f(x)=x**2 + 10
gnuplot> g(x) = -x**2 -10
gnuplot> plot f(x) ls 1, g(x) ls 2
```

plot - Styles de traçage

Exemple



```
gnuplot> set style function Linespoints
gnuplot> f(x)=x**2 + 10
gnuplot> g(x) = -x**2 -10
gnuplot> plot f(x) , g(x)
```



```
gnuplot> set style function Linespoints
gnuplot> f(x)=x**2 + 10
gnuplot> g(x) = -x**2 -10
gnuplot> set ls 1 lt 4
gnuplot> set ls 2 lt 7
gnuplot> set samples 10 # nombre de points
gnuplot> plot f(x) ls 1 , g(x) ls 2
```

Personnaliser votre graphique

commandes les plus fréquentes

commande	signification
<code>gnuplot> set title 'LMA'</code> <code>help title</code>	crée le titre LMA au-dessus du graphe (voir l'aide de la commande)
<code>gnuplot> set xlabel 'Drag (N)'</code> <code>help xlabel</code>	place l'étiquette Drag (N) sur l'axe_x (voir l'aide de la commande)
<code>gnuplot> set ylabel 'Lift (N)'</code> <code>help ylabel</code>	place l'étiquette Lift (N) sur l'axe_y (voir l'aide de la commande)
<code>gnuplot> set xrange [-5:5]</code> <code>help xrange</code>	Fixe l'intervalle de l'abscisse (axe_x) (voir l'aide de la commande)
<code>gnuplot> set yrange [-1.:10]</code> <code>help yrange</code>	Fixe l'intervalle de l'ordonnée (axe_y) (voir l'aide de la commande)
<code>gnuplot> set autoscale</code> <code>help autoscale</code>	Ajuste automatiquement les intervalles des axes (voir l'aide de la commande)
<code>gnuplot> set key 2,45</code> <code>help key</code>	affiche la légende aux coordonnées (x=2,y=45) (voir l'aide de la commande)
<code>set key bottom right</code> <code>help key</code>	affiche la légende en bas à droite sur graph (voir l'aide de la commande)
<code>set lmargin at screen 0.1</code> <code>help margins</code>	règle la marge à gauche à 10% de la taille totale du graphe (voir l'aide de la commande)

Personnaliser votre graphique

commandes les plus fréquentes

commande	signification
<code>gnuplot> set mxtics 4</code> <code>help mxtics</code>	utilise 4 marques mineurs entre les marques majeurs sur l'axe_x. (voir l'aide de la commande)
<code>gnuplot> set mytics 4</code> <code>help mytics</code>	introduit 4 marques mineurs entre les marques majeurs sur l'axe_y. (voir l'aide de la commande)
<code>gnuplot> set terminal pngcairo enhanced color</code> <code>help terminal</code>	Choisi le terminal pngcairo comme terminal de sortie (voir l'aide de la commande)
<code>gnuplot> set output 'Ux1.png'</code> <code>help output</code>	Crée le fichier de sauvegarde du graph dans le répertoire courant. (voir l'aide de la commande)

Reference : cours CFD et Logiciel Dr ABED Bouabdelah, Dr HAMEL
Mohamed