

Les commandes utilisées dans ce TP vont faire apparaître un certain nombre de fenêtres graphiques. Si vous ne les fermez pas au fur et à mesure, tous les tracés se superposeront dans la même fenêtre.

1 Tracer les termes d'une suite

1.1 Avec bar

La commande `bar` prend en argument un tableau et renvoie un diagramme en bâton représentant ses valeurs, avec en abscisse les numéros des cases du tableau.

Exercice 1. Soit u la suite définie par $u_1 = 0,9$ et $\forall n \in \mathbb{N}^*, u_{n+1} = u_n^2$. Construire un tableau U contenant les 8 premiers termes de la suite et les représenter graphiquement avec `bar`.

1.2 Avec plot

La syntaxe de `bar` est simple, mais n'offre qu'un choix limité de tracés. La fonction `plot` est plus complexe, mais propose aussi plus de flexibilité (en particulier pour la gestion de l'axe des abscisses). Sa syntaxe est la suivante :

```
plot(x,y,"+")
```

x et y doivent être deux vecteurs de même taille, représentant respectivement les abscisses et les ordonnées des points à tracer. Le dernier argument, "+", désigne le type de points que l'on veut pour le tracé. Il est possible de le remplacer par d'autres, en particulier "o" ou "*", pour varier les styles. On peut aussi faire varier les couleurs : "r+" pour du rouge, "g+" pour du vert, ...

Exercice 2. Réutiliser le tableau U de l'exercice précédent et représenter graphiquement les valeurs qu'il contient avec `plot`.

2 Tracer la courbe d'une fonction

Pour tracer une courbe $y = f(x)$, on se contente de tracer suffisamment de points du type $(x, f(x))$ sur un graphe et de les relier pour qu'ils donnent l'impression d'une courbe.

2.1 Choix des abscisses

Il existe plusieurs méthodes pour construire un tableau de valeurs pour les abscisses. Avec les techniques des TP précédents, on sait par exemple que `a:h:b` crée un tableau contenant des valeurs réparties régulièrement (avec un pas de h) dans l'intervalle $[a, b]$.

Une autre possibilité est d'utiliser la commande `linspace(a,b,n)`, qui renvoie un tableau contenant n valeurs réparties uniformément entre a et b . C'est souvent la syntaxe privilégiée, parce qu'elle permet une bonne maîtrise du nombre de valeurs à tracer.

2.2 Syntaxes de plot

Une fois le choix des abscisse effectué, c'est la fonction `plot` qui permet de tracer la courbe. Elle prend deux arguments : le tableau des abscisses et la fonction à tracer ou le tableau des ordonnées.

Exemple 1. Pour tracer la fonction cos sur $[-2, 5]$, on utilise par exemple :

```
x=linspace(-2,5,50); plot(x,cos)
```

Le choix de 50 points est arbitraire ici. Il faut utiliser une valeur assez petite pour que la courbe se trace rapidement, mais assez grande pour donner l'impression d'une courbe « lisse ».

Exercice 3. Tracer la fonction cosinus sur $[-2, 5]$ pour les abscisses suivants : `linspace(-2,5,10)`, `linspace(-2,5,20)`, `linspace(-2,5,50)`, `linspace(-2,5,500)`. Que peut-on observer ?

Exercice 4. On considère la fonction f définie sur $[-4, 4]$ par $x \mapsto \frac{x^3 - x - 3}{2x^2 + 1}$. Tracer sur un même graphique la courbe de cette fonction et son asymptote d'équation $y = \frac{x}{2}$.

2.3 Choix de la couleur, du type de tracé et propriétés des axes

Ces syntaxes ne sont pas au programme, donc pas exigibles, mais elle vous aideront à améliorer la lisibilité de vos graphiques.

En ajoutant un troisième argument à `plot`, il est possible de choisir la couleur de la courbe : "g" pour vert, "c" pour cyan, "y" pour jaune, "m" pour magenta, "r" pour rouge.

Exemple 2. Pour tracer la courbe en rouge, on écrira

```
x=linspace(-2,5,50); plot(x,cos,"r")
```

On peut aussi choisir le type de ligne : "--" (tirets), "-." (mélange de tirets et pointillés), ...

Exemple 3. Pour tracer la courbe en rouge et avec des tirets, on écrira

```
x=linspace(-2,5,50); plot(x,cos,"r--")
```

Exercice 5. Tracer sur un même graphe les courbes de cosinus (en rouge) et sinus (en bleu et tirets) sur $[-5, 5]$.

Il est possible d'ajouter une légende, un titre, de donner des noms aux axes :

- La commande `legend` affiche une légende.
- La commande `xtitle` donne un titre au graphique.
- Les commandes `xlabel` et `ylabel` donnent des noms aux axes

Exemple 4. On peut par exemple écrire :

```
x=linspace(-2,5,50); plot(x,cos)
xtitle("La fonction")
legend("cosinus")
```

2.4 Tracer plusieurs courbes en plusieurs graphiques

Ces syntaxes ne sont pas au programme, donc pas exigibles, mais elle vous aideront à améliorer la lisibilité de vos graphiques.

Si l'on souhaite afficher plusieurs graphiques dans une même fenêtre, il faut utiliser la commande `subplot(m,n,p)`, qui divise la page en m lignes et n colonnes de surface identique et trace le graphique dans l'élément p . Les éléments sont numérotés de la gauche vers la droite et du haut vers le bas.

Exercice 6. Expliquer ce que font les commandes suivantes :

```
x=linspace(-4,4,50);
subplot(1,2,1), plot(x,cos,"r")
subplot(1,2,2), plot(x,sin,"g+")
```

2.5 Une alternative : `plot2d`

On peut remplacer la fonction `plot` par la fonction `plot2d` pour tracer une courbe. Elle prend en argument deux tableaux et offre un rendu assez similaire à celui de `plot`.