

第1章 基本操作

1.1 最初の一葉 (FirstLeaf): 文法とヘルプとプロット

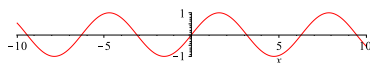
[[解説]]

入力領域と注意点 (ShiftEnter)

Maple を起動すると赤いプロンプトがとまっている。ここに命令 (コマンド) を打ち込んで Maple の計算部に『こちらの意志』を伝えて動作させる。例えば,

```
> plot(sin(x),x);
```

と入力し, `enter` を入力してみよ, \sin 関数がプロットされる。



1. 赤い領域のどこにカーソルがあっても `enter` を入れれば, そのブロックごと Maple に命令として渡される。テキストでは `enter` を省略している。
2. テキストの修正は普通のワープロソフトと同じ。
3. 命令の入力ではなく, **改行だけをいれたいときは shift+enter を入れる。**
4. 命令は, enter を入れた順に解釈されるのであって, テキストの上下とは関係ない。

命令コマンドの基本形 (command();)

命令コマンドは全て次のような構造を取る。

```
> command(引数 1, 引数 2,...);
```

あるいは

```
> command(引数 1, オプション 1, オプション 2,...);
```

となる。

1. () の中の引数やオプションの間はコンマで区切る。
2. 最後の; (セミコロン) は次のコマンドとの区切り記号。
3. セミコロン (;) をコロン (:) に替えると Maple からの返答が出力されなくなるが, Maple への入力が行われている。
4. C 言語などの手続き型プログラミング言語の標準的なフォーマットと同じ。

命令コマンドを, 英語の命令文と解釈してもよい。たとえば,

$\sin(x)$ を x について 0 から π まで plot せよ。

という日本語を英語に訳すと,

plot sin(x) with x from 0 to Pi.

となる. この英語を Maple 語に訳して

```
> plot(sin(x),x=0..Pi);
```

となったとみなせる. 英語文法の Verb (動詞), Object (目的語) を当てはめると, Maple への命令は,

```
> Verb(Object, その他の修飾);
```

である. 英文でピリオドを忘れるなど中学時代に言われたのと同じく, Maple でセミコロンを忘れぬように.

ヘルプ (?)

ヘルプは少し違った構文で, 例えば先程のコマンド plot のヘルプを参照するときには,

```
> ?plot;
```

である.

ヘルプ画面は, 左側に操作アイコン, 検索ウィンドウ, 関連リストが表示され, 右側にヘルプの本文がある. 本文は, 簡単な意味と使い方, 説明, 例, 参照で構成される. ほとんどが日本語に訳されているが, 古いテキストやあまり使わないコマンドは英語のまま. 英語が分からなくても例を参考にすればだいたい予測できる. というより, 日本語訳を読んでも初めはチンプンカンプン. Maple コマンドのコンセプトに慣れるまでは使用例をまねるのが一番の早道でしょう.

間違い (Error)

いくつかの典型的な間違い. まずは, 左右の括弧の数が合っていないとき.

```
> plot(sin(x),x=-Pi..Pi);
```

Error, ';' unexpected

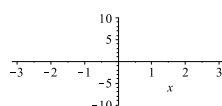
正しくは,

```
> plot(sin(x),x=-Pi..Pi);
```

です. 関数の中に変数が残ったまま plot しようとしたとき.

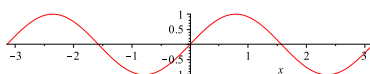
```
> plot(sin(a*x),x=-Pi..Pi);
```

Warning, unable to evaluate the function to numeric values in the region; see the plotting command's help page to ensure the calling sequence is correct



何も表示されない. x に 1 を代入しても, $\sin(a*x)$ からは数値ではなく記号で答えが返って来ている. plot は関数が数値を返したときしか表示できない. 以下のように, 変数 a のかわりに数値を入れる.

```
> plot(sin(2*x),x=-Pi..Pi);
```



[[[課題]]]

plot に関する課題

1. plot に関する help を開けよ.
2. 例をいくつか copy して実行せよ. 円をパラメータプロットする方法を確認せよ.
3. 2つの関数, $\sin(x)$ と $\cos(x)$, を $x=-\text{Pi}..\text{Pi}$ で同時にプロットせよ.
4. 上記の2つの関数の表示で, オプションに `color=[red,blue]` および `style=[point,line]` を加えて, 結果を観察せよ. `plot[options]` に関する help を開け, そのほかのオプションを試してみよ.

plot3d に関する課題

1. x,y の2次元平面を定義域とする関数のプロットには, 3次元で表示するコマンド `plot3d` が必要となる. 2変数関数 $\sin(x)*\cos(y)$ を $x=-\text{Pi}..\text{Pi}$, $y=-\text{Pi}..\text{Pi}$ でプロットせよ.
2. プロットをつまんで回してみよ. また, メニューバーにある表面あるいは軸のアイコンを操作して, 等高線にしたり, 軸を変更してみよ.

[[解答例]]

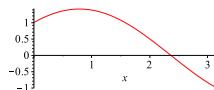
plot

1.

```
> ?plot;
```

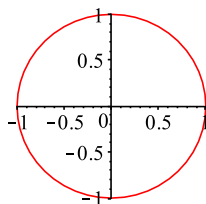


```
> plot(cos(x)+sin(x), x = 0 .. Pi);
```



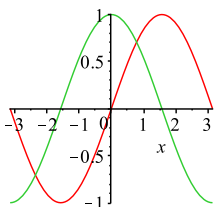
2.

```
> plot([sin(t), cos(t), t = -Pi .. Pi]);
```



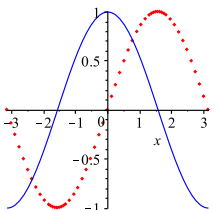
3.

```
> plot([sin(x), cos(x)], x=-Pi..Pi);
```



4.

```
> plot([sin(x), cos(x)], x=-Pi..Pi, color=[red,blue], style=[point,line]);
```



plot3d

plot 図に対するいくつかの操作は, plot を選んだときに window の上部に表示されるリボンのアイコンにあります. 同じ操作は, plot3d に与える options によっても可能です. 以下でいくつかの option を示します. 詳しくは `?plot3d[option]`; を参照.

1. `> plot3d(sin(x)*cos(y),x=-Pi..Pi,y=-Pi..Pi);}`



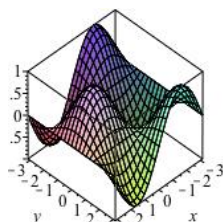
2. 等高線は

`> plot3d(sin(x)*cos(y),x=-Pi..Pi,y=-Pi..Pi,style=contour);`



軸の変更は

`> plot3d(sin(x)*cos(y),x=-Pi..Pi,y=-Pi..Pi,axes=boxed);`



3. デフォルトの角度も plot3d の option で変更することが可能です.

`> plot3d(sin(x)*cos(y),x=-Pi..Pi,y=-Pi..Pi,orientation=[45,80]);`

