

情報科学演習 第13回

数式処理系 Maple 入門

1 はじめに

今回は数式処理系 Maple を紹介します。前回、リモートで情報基盤統括センターのマシンを使用しましたが、それを利用して、リモートから maple を使います。maple 自身、グラフィカルな環境でも利用でき、その場合には関数のグラフを描画させるなどの機能もあります。しかし、今回はそのような環境が利用できないので、純粋に数式の計算法を述べます。

数式処理とは、数学で行う計算、即ち、式の展開、因数分解、微分、積分、方程式を解く、逆行列を求める、固有値計算等を数式のままコンピュータでする事を指します(数式を美しく印刷する作業を数式処理と言う人もいますが、通常はこちらは組版(くみはん)処理といいます)。

現在のコンピュータが開発(von Neumann 型の完成が 1950 年前後)からあまり間をおかず(1960 年台半ば)、数式処理システムの開発が始まっています。もっとも早い時期の AI(Artificial Intelligence, 人工知能)のひとつであるとも言えます。

実際 Maple, Mathematica, Maxima(sage, Macsyma), Reduce 等の数式処理ソフトは、大学初年級以上の数学を処理する能力を持っており、数学教育で用いるソフトとして便利ですし、研究上の実験ソフトとしても役に立ちます。

今回は、大学に導入されている Maple(これは有料のソフトウェア)をリモートで使ってみます。このテキストの最後に、無料の数式処理ソフトである Maxima の導入法を書きました。

2 Maple とは

Maple は、Canada の Waterloo 大学で開発された数式処理ソフトです。Maple は、沖縄では見ませんが、楓という木の事です。Canada 国旗の中央部には楓の葉がデザインされています¹。現在、Waterloo Maple Inc. が販売しております。サポートされている OS は、MacOS, Windows, Solaris, Linux (x86) 等です。

Maple の開発は、1980 年頃に始まったようです。ただし、数式処理が PC で実行可能になって来たのは 1990 年頃からで、それまでは、Workstation と呼ばれる少し大きめのシステムでしか、動作しませんでした(現在では、Workstation と PC の差はほとんどなくなっている.)。

このテキストでの約束: このテキストでは、> で始まっている行は、Maple の入力です。

Maple は一般ユーザ版は 20 万円以上しますが、学生版 (Student Edition) は、12000 円 (税抜き?) で購入出来るようです。

¹個人的には、パンケーキのシロップは、メイプルシロップ。

3 Maple の起動

前回同様、コマンドラインで `cc.u-ryukyu.ac.jp` に `ssh` を用いて接続します。コマンドプロンプト (Windows 10), ターミナル (ターミナル.App)(Mac, Windows 11) で、次を入力してください (e223lxx の部分は、自分の login 名を入力)。

```
ssh cc.u-ryukyu.ac.jp -l e2231xx
```

iPad では、Koder, SwiftTermApp のどちらを使っても、今回は大丈夫です。

前回の講義を実行している人は、password 入力になりますが、飛ばした人は、接続を続行するか (認証) に対する質問が出ますので、yes とタイプしてエンターを押すと、password 入力となります。

例によって、password は画面表示されません。password は 3 回間違えると、接続が自動的に切れます。その際には、上を入力して再接続してください。上の ssh で、-1 に続く部分は、login 名ですが、ここを間違えた場合は、CTRL-c(Control キーを押しながら c を押す) で、ssh の実行が中止できます。

cc.u-ryukyu.ac.jp に接続を完了しますと、コマンドプロンプトが現れますので, maple とタイプして ENTER を押します.

```
bash-4.4$ maple
```

次のような maple からの起動メッセージが表示されます。左側のアスキーアート (アスキー文字で絵を描いたもの) は, Canada の国旗の中央にある楓の葉を真似たものです。Windows の場合は, \ が ¥ で表示されると思います。

```

|\~/|      Maple 2020 (X86 64 LINUX)
._|\||  |/_|. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2020
 \  MAPLE  / All rights reserved. Maple is a trademark of
<____ ____> Waterloo Maple Inc.
      |      Type ? for help.
>

```

この状態で, maple を用いた計算が対話的に実行できます.

3.1 簡単な計算

Maple の四則演算の記号はそれぞれ, $+$, $-$, $*$, $/$ です。Maple では冪乗と階乗が定義されており, それぞれ, $\wedge$ と $!$ を用います。Maple では, 計算式の最後をセミコロン ; で終って ENTER キーを押すと, 計算式の評価結果が出力されます。ENTER キーだけでは単なる改行となります。セミコロンを忘れて ENTER キーを押した場合, 次の行で改めてセミコロンを押してエンターを押せば, 式は計算されます。入力間違いがあって error が表示された場合も, 気にせずに使い続けて下さい。式の中の括弧 () は数学と同じ意味になります。四則演算の優先順位も数学と一致します。次ページの先頭部分を実行してみてください。

```
> 1 + 2;
> 10/3 + 2;
> 10/3.0;
> 2^10;
> 50!;
> (a+a-b)*c/d;
> a^2 + a;
```

有理数の扱い, 文字式の扱いが数学と一致します. 小数が式に含まれていれば, 自動的に小数扱いされます.

3.2 様々な定数, 手続き (関数)

Maple では, 数式処理のための手続きが 2700 以上定義されています. それらを全部解説する事は不可能ですので, ここではその一例をあげます. これらの例の中に初等関数が用いられていますが, それらの意味は容易に類推できると思いますので, これについての解説はいちいちしません.

数式処理に欠かせない定数 (円周率等) が既に定義されています. 円周率は, Pi という記号を使います. 関数に値を代入する時には, 必ず括弧 () が必要で, しかも数式計算上の括弧は, これ以外には使えません. { }, [] は別の意味になります.

```
> Pi;
> cos(Pi/4);
> tan(Pi/2);
> arctan(-infinity);
```

$\sqrt{2}$ がそのまま出て来る事, tan の不定値に対するエラーメッセージに注意して下さい. arctan は逆正接関数, infinity は無限大の事です. 最後の答は, 極限值を出力しています.

Maple では, 有理数, 冪根, 円周率等の定数は, そのまま出力されます. 上の逆正接関数の計算でもそうですし, 例えば, $1 + \frac{1}{2^2} + \frac{1}{3^2} + \cdots = \sum_{n=1}^{\infty} \frac{1}{n^2} = \zeta(2) = \frac{\pi^2}{6}$ も次で計算させると, 円周率を使った答が出ます.

```
> sum(1/n^2, n=1..infinity);
```

これを小数へ変換するには, evalf(evaluate float) という手続きを用います.

```
> evalf(Pi^2/6);
```

Maple では, 非常に正確な数値計算ができます. 例えば, $e^{\pi\sqrt{163}} - 744 - 640320^3$ を C 言語の数学関数で計算しますと -480 という答を得ます. Maple を使うと, C 言語の計算がとんでもない誤差を含む事がわかります. 起動時では, 浮動小数点の仮数部は 10 桁に設定されており, そのままですと, この結果は真の値の約 40 倍という答になります. そのため仮数部の桁数を事前に設定します. 仮数部の桁数は, Maple のシステム変数 Digits に格納されていますから, この値を変更します. Maple では変数への代入に := を用います.

```
> Digits:=50;
> evalf(exp(Pi*sqrt(163)));
> evalf(exp(Pi*sqrt(163))-744-640320^3);
```

$e^{\sqrt{163}\pi}$ の計算結果で 9 が 12 個続く部分がありますが, 先頭の 9 と 2 番目の 9 の間に小数点があります. C 言語のライブラリを用いた計算が, 真の値の 6×10^{14} 倍以上の値になっている事がわかります. (exp は exponential(指数関数) の sqrt は square root(平方根) の略)

課題 (難): $e^{\sqrt{163}\pi}$ の値が整数に近い理由を調べよ. 菅にこれをきちんと説明できる人は, 卒業まで私が担当する科目の全ての単位を (授業登録すれば) A であげます. (同じ理由で, $e^{\sqrt{67}\pi}$ も整数に近い.)

文字式の展開, 因数分解も可能です.

```
> expand((x+y)^5);  
> factor(a^8-b^8);
```

次の問題は, 2000 年の琉球大学入学試験問題前期日程数学甲の 1 です.

1. 関数 $\frac{x}{\sqrt{1+x^2}}$ を微分せよ.
2. 不定積分 $\int x\sqrt{x^2+2} dx$ を求めよ.
3. 定積分 $\int_{\frac{1}{e}}^{2e} x^3 \log x dx$ を求めよ.
4. $\lim_{x \rightarrow 0} \frac{\sqrt{2x+1} - 1 - x}{x^2}$ を求めよ.

このような単純な計算は, Maple は得意です. (試験では, 答だけを書いても満点にはならないと思います.) Maple V では, 直前の結果を % で参照できます.

```
> diff(x/sqrt(1+x^2), x);  
> simplify(%);  
> int(x*sqrt(x^2+2), x);  
> int(x^3*log(x), x=exp(-1)..2*exp(1));  
> limit((sqrt(2*x+1)-1-x)/x^2, x=0);
```

方程式 $f(x) = 0$ の解を求める様々な方法も Maple には用意されています. $f(x)$ が 4 次以下の多項式なら, この方程式には代数的な解法が存在する事が知られています. (3 年の代数学 I・II で勉強する予定です.) Maple はこれらの解法を知っており, solve という手続きになっています. 次を実行してみてください. これらの解には複素数が含まれますが, Maple では虚数単位は大文字の I で表示されます.

```
> solve(x^3+1, x);  
> solve(x^3+3*x+1, x);
```

5 次以上の方程式には, 代数的な解法が一般には存在しない事が知られています (代数的という制限を外せば, 別な解法はあります). 次を実行してみてください.

```
> solve(x^5+x^2+1, x);
```

$\text{RootOf}(_Z^5 + _Z^2 + 1)$ という解が出て来ます。もちろん、これは単なるトートロジーに過ぎないのですが、Maple は代数的数を扱えるので、この解 (代数的数) を用いた計算が今後記号的に可能です。

代数的な解法がある場合でも、その解法が複雑な場合には、残念ながら代数的な解を出力しません。例えば、1 の 7 乗根を計算させようとしても、de Moivre (ド モアブル) の公式から出てくる解が単純に出力されるだけです。

```
> solve(x^7-1, x);
```

上で述べたように Maple では、代数的数が扱えます。これを利用すると、 $x^7 - 1 = 0$ の代数的な解も求める事ができます。 $x^7 - 1 = (x - 1)(x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$ ですが、積の右側の 6 次式は $\sqrt{-7}$ を使うと、2 つの 3 次式の積に因数分解されます。この様な因数分解は、付け加える数を factor の第 2 引数に加える事で可能です。

```
> factor(x^6+x^5+x^4+x^3+x^2+x+1, (-7)^(1/2));
```

従って、1 の複素 7 乗根は、次のベキ根だけを利用して記述すると、2 つ方程式の解全体です。

```
> solve(2*x^3+x^2-I*sqrt(7)*x^2-x-I*sqrt(7)*x-2,x);
```

```
> solve(2*x^3+x^2+I*sqrt(7)*x^2-x+I*sqrt(7)*x-2,x);
```

$f(x)$ が 5 次以上の多項式や、多項式以外の場合には、一般的な解法が存在しません。このような方程式の近似解を数値的に計算する方法も Maple は知っています。上の方程式の数値解も、次で計算してくれます。

```
> fsolve(x^5+x^2+1, x);
```

3.2.1 パッケージ (package) を利用する

これまでの計算では、特別な事をせずともそれを実行しますが、行列の計算などはそのままでは実行してくれず、それを計算するためのパッケージ (ライブラリ) を呼び出す必要があります。ここでは、線形代数学のパッケージを呼び出して、そこで定義されている計算をしてみます。

パッケージを読み込むには with という手続きを使います。線形代数学のパッケージ名は linalg となっておりますので、次のように入力します。

```
> with(linalg);
```

この時に出力されるのが、linalg のパッケージで定義されている手続き名です。一例をあげますと、次のようになります。(matrix: 行列, trace: 行列のトレース, det: determinant(行列式), inverse: 逆行列, eigenvalue: 固有値)

```
> B:= matrix(2,2);
```

```
> trace(B);
```

```
> det(B);
```

```
> inverse(B);
```

```
> eigenvalues(B);
```

どのようなパッケージが利用できるかは、手続き help に引数 package をいれて利用する事でわかります。ただし、この環境では出てくるメッセージは英語です。

```
> help(packages);
```

多量のメッセージが現れ、1 画面には表示されません。最下行に `-- More: packages --` (色が反転している) と出て来たら、スペースキーを押すと、1 画面進みます。Help メッセージの表示の終了は、q キーを押します。

3.3 Help

上にあるように、Maple は Help 機能があり、英語が読めれば割と有用です。グラフィカルな環境では、Help も日本語訳が付いているようです。使い方は、

```
> ? キーワード
```

あるいは

```
> help(キーワード);
```

とすると、それに対する Help が表示されます。例えば、

```
> help(diff);
```

とすると、微分するコマンド `diff` の使い方が表示できます。検索すべきキーワードが何であるかというのは、他の数学の講義から学習してください。

```
> help(help);
```

Maple の終了

Maple の利用を終了するには、次のように入力します。

```
> quit;
```

`cc.u-ryukyu.ac.jp` との接続を切るには、Maple を終了した後、

```
bash-4.4$ logout
```

で、logout します。logout せずにターミナルなりコマンドプロンプトを終了しても、特に問題は起きません。

このような数学用ソフトやプログラミング言語の記述、あるいはそのエラーメッセージは、ほとんどの場合英語になります。それは、これらが世界中で開発されており、そのときの共通言語が英語になってしまったからです。そのような英語のメッセージを日本語訳は、なされていないのが多くあります。従って、英語に対するアレルギーをなくすとともに、そのメッセージを読む訓練をしておくことは、将来の役に立ちます。文学のような難しい表現はありませんので、はやいうちに慣れて下さい。また、微積分や線形代数の教科書の問題を Maple で解いてみて下さい (解答にそこそこ間違いがあったりして...).

2002 年の計算機言語 I で Maple を用いたプログラミングの講義を行いました。そのときのテキストが (L^AT_EX ファイルですが)、次の場所にあります。今回のテキストは、そこにある 1.tex を改訂した物です。使い方をより知りたい方は、参考にして下さい。Maple の version が当時から大分上がっていますが、内容は今でもほとんど通用するはずで、来年の計算機概論 I でも、もう一度 Maple を取り上げる予定です。

<ftp://ftp.math.u-ryukyu.ac.jp/pub/gengo/2002/>

参考文献

- [1] B. W. Char 他, サイバネットシステム訳, はじめての Maple, 1998, シュプリンガーフェアラーク東京
- [2] B. W. Char 他, サイバネットシステム訳, よくわかる Maple, 1998, シュプリンガーフェアラーク東京
- [3] K. M. Heal, M. Hansen, K. Rickard 著, 示野信一他訳, Maple V Release 5 ラーニングガイド, 1999, シュプリンガーフェアラーク東京
- [4] 示野信一著, Maple V で見る数学ワールド, 1999, シュプリンガーフェアラーク東京

Maple の使い方の参考書です。どれも古い本なので今とは少し使い方が異なる部分もありますが, そのときには Help で調べて下さい。また, 現在では, これらの本は流通していないと思いますので, 図書館等で借りて呼んで下さい。

上記の本以外に, Maple Programming Guide(英語) が, 電子書籍として無料で読めるようです。

4 Maxima の導入

無料の数式処理系で古くからある有名なものとして, Maxima(macsyma) とそれを拡張した, Sage があります。Maple からは, 少しだけ機能が劣るようですが, これらをネットから取ってきて使うという手もあります。Maxima は比較的汎用な数式処理ソフトですが, Maxima 以外にも数学計算のための無料ソフトが沢山あり, これらのうち, 有用なものをまとめて使えるようにしたものが, Sage です。Sage は python というプログラミング言語を制御用の言語として用いて, 様々な数学ソフトを一括して利用できるようにしているものですが, 少し使ってみた感じでは, うまく動かない部分も多いようです(現時点では無理に纏めるより, 個々に使った方が良いでしょう)。

ここでは, Maxima をグラフィカルに使えるようにした wxMaxima のインストール方法を OS 別に解説します。

使い方は以下の文献を参考に自習して下さい。

入門的なもの

http://www.k-techlabo.org/www_maxima/maxima_main.pdf

<https://home.hirosaki-u.ac.jp/heroic-2020/wp-content/uploads/sites/25/wxmaxima.pdf>

本格的な読み物 (分量多い)

<http://fe.math.kobe-u.ac.jp/MathLibre-doc/ponpoko/MaximaBook.pdf>

Windows

Windows は下の場所から最新版をダウンロードして、ダブルクリックするだけです.

<https://sourceforge.net/projects/maxima/files/Maxima-Windows/>

最近の PC だと 64 bit 版が正解だと思います.

Mac

上で示した SourceForge のサイトには, Mac 用は, ソースコードだけがあるようです. まず, MacPorts というシステムを導入して, その後 SourceForge にあるソースコードをコンパイルして実行ファイルを作る形になるようです. 残念ながら, 私は, MacPorts を利用していないので, この方法の解説はできません.

上のことが面倒な場合, Home Brew を利用してインストールする方法もあります. まず, 下の場所から, Home Brew を導入します.

https://brew.sh/index_ja

その後, ターミナルで,

```
brew install wxmaxima
```

を実行すると, wxMaxima がネットから自動的にインストールされます. この場合, wxMaxima を利用するには, ターミナルから, 次のようにタイプして ENTER キーを押します.

```
wxmaxima
```

iPad

iPad では, Maxima は動作しないようです. 数式処理ソフトとしては, SageMath(Sage) が動作するようですので, AppStore を探してみてください. SageMath は, 利用したことがないので, 解説はできません.

MathLibre

上に述べた Maxima や, これ以外の数学用の無料ソフトをパッケージ化し Linux 上にまとめたものとして, MathLibre (<http://www.mathlibre.org/index-ja.html>) というものが日本で開発されています. USB メモリから起動可能で, USB 3.0 ならそこそこ実用的なのですが, Intel 系の CPU でしか動作しない (最近の Mac はダメ), PC の BIOS(Basic Input Output System) の Secure Boot を回避するのが面倒などの問題があります.

MathLibre に数理の授業で必要になりそうなソフトを追加して, 16GB の USB メモリで使えるシステムを作りましたので, 試してみたい人は, 私まで連絡を下さい.