

数式処理—Mapleの全容と鉄則—

Copyright ©2006 by Shigeto R. Nishitani

Mapleは数学あるいはプログラミングを補助する道具として非常に便利である。使いこなせば、数式の変形や簡単なプログラム開発が、体感として10倍高速になる。しかし、便利だけに覚えるのも難しそうに感じるかもしれない。確かに、ワープロや表計算ソフトとは違い、いくつものコマンドを知っておく必要がある。どこまでを知るあるいは覚える必要があるのかに不安を感じるのは当然である。私の日頃使っているコマンドを項目に分けて整理すると以下のように分類できる。ここで、最初に必要になるのは、1-5までの初・中級の項目だけである。初級は最低限必要な知識を、中級はこれだけ知っていればどんな難問もこなせる知識を、それぞれ意味している。だいたい5コマ程度の演習で基本操作は身に付く。本演習で提供するチャート式テキストによって、コマンドをまったく「覚える必要なく、使える」ことを意図している。

Mapleコマンドの分類とテキストで扱う内容

		初級	中級	発展・応用課題
1	数式処理 (symbolic computing)	基本操作 ヘルプ 代入(四則演算) 関数	コマンド 鉄則 複雑な積分 数式変形の基本問題 (微積, 線形代数の発展課題)	Thermal expansion, Einstein, Tunnel effect
2	plot	プロット	plots, plottools, animation	RSA, sort
3	微積 (Calculus)	ユーザー定義関数, 方程式の解, 微分・積分	微積のちょっとひねった問題をここで、例えば、2重積分, 全微分, 接線,	FFT
4	線形代数 (Linear Algebra)	Vector&Matrix, Inner&Outer product	Transpose&Adjoint, Det&Inverse, Eigenvectors,	PageRank, LargeMatrixInverse, 非線形最小2乗
5	プログラム		printf,	Traveling Salesman,

	(Programming)	if, for-loop, Array, (list構造と両方必要, さらにsetも) proc	Tower of Hanoi, 8-Queens, AB alloy
6	ファイルの入出力	file, in/out, filter	linear-fitting
7	数値計算	代数方程式, 誤差, 逆行列(LU分解), 固有値, 補間と数値積分, 線形最小2乗法, 非線形最小2乗法, CG, SA, FFT,	
8 ...	微分方程式 その他のパッケージ	dsolve	こーひー, 予測子-修正子, ルンゲクッタ

鉄則

Mapleをはじめとする数式処理ソフトあるいはプログラミングの習得にあたって初心者がつまづく共通の過ちを回避する鉄則がある。

鉄則0：restartから始める

続けて入力すると前の入力が生きている。違う問題へ移るときや、もう一度入力をし直すときには、restartを入力して初期状態からはじめる。入力した順番が狂っている場合もある。頭から順にenterをやり直す。

鉄則1：出力してみる(プログラミングでも)

多くのテキストではページ数の関係で出力を抑制しているが、初心者が問題を解いていく段階ではデータやグラフをできるだけ多く出力する。最後のコロンをセミコロンに変える、あるいは途中でprint文を入れる。

鉄則2：関数に値を代入してみる

数値が返ってくるべき時に変数があればどこかで入力をミスっている。

鉄則3：内側から順に入力する(プログラミングでも)

長い入力やfor-loopでは内側から順に何をしているか確認しながら打ち込む。括弧が合わなかったり、読み飛ばすというエラーが回避できる。