

## 数式処理—鉄則と具体例(1)—

Copyright ©2006 by Shigeto R. Nishitani

どうしても解かなければならない課題を前にコマンドリファレンスのあちこちを参照しながら解いていくのが数式処理を修得する最速法である。とびかかる前にちょっとした共通のコツがある。それをここでは示す。数式処理ソフトでの数式処理とは、数式処理ソフトが『自動的にやって』くれるのではなく、実際に紙と鉛筆で解いていく手順を数式処理ソフトに『やらせる』ことであることを肝に銘じよ。

### 鉄則

Mapleをはじめとする数式処理ソフト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 をかける

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returnをやり直す。

#### 鉄則1：出力してみる

多くのテキストではページ数の関係で出力を抑止しているが、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。plotで以下のようなエラーが出た場合にチェック。

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

```
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
```

```
Error, empty plot
```

#### 鉄則3：内側から順に入力する

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、読み飛ばしていたりというエラーが回避できる。

### 具体例：無限積分

以下に示す積分を実行せよ。

$$\int_{-\infty}^{\infty} x e^{-\beta c x^2} (1 + \beta g x^3) dx$$

最新版のMapleでは改良が施されていて、このような複雑な積分も一発で求まる。

```
> f1:=unapply(x*exp(-beta*c*x^2)*(1+beta*g*x^3),x);
```

$$f1 := x \rightarrow x e^{-\beta c x^2} (1 + \beta g x^3) \quad (1.2.1)$$

```
> int(f1(x),x=-infinity..infinity);
```

$$\begin{cases} \frac{3}{4} \frac{g \sqrt{\pi}}{\beta c^2 \sqrt{\beta c}} & \text{csgn}(\beta c) = 1 \\ \infty & \text{otherwise} \end{cases} \quad (1.2.2)$$

ここでは、 $\beta c$ が正の場合( $\text{csgn}(\beta c)=1$ )とそれ以外の場合( $\text{otherwise}$ )に分けて答えを返している。しかしこのような意図したきれいな結果をいつもMapleが返してくれるとは限らない。これだけしか知らないと、なにかうまくいかないときにお手上げになってしまう。このようなきれいで簡単な結果に行き着く前の、裏でおこなういくつかの予備計算を省略せずに示そう。

先ず鉄則0にしたがってrestartをかけ、関数を定義する。

```
> restart;
```

```
f1:=unapply(x*exp(-beta*c*x^2)*(1+beta*g*x^3),x);
```

$$f1 := x \rightarrow x e^{-\beta c x^2} (1 + \beta g x^3) \quad (1.2.3)$$

次には鉄則1にしたがって積分する前にどのような関数かプロットしてみる。そのままplotへ投げると怒られる。

```
> plot(f1(x),x=-10..10);
```

```
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
```

```
Error, empty plot
```

これは鉄則2にあるとおり、数値を代入すれば、beta,c,gなどのパラメータの値が入っていないためとわかる。

```
> f1(10);
```

$$10 e^{-100\beta c} (1 + 1000\beta g) \quad (1.2.4)$$

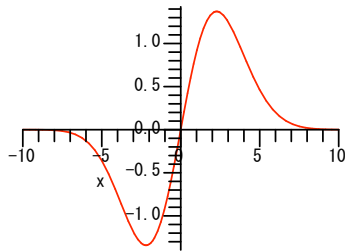
適当に値を代入する。

```
> c:=1; g:=0.01; beta:=0.1;
```

$$\begin{aligned} c &:= 1 \\ g &:= 0.01 \\ \beta &:= 0.1 \end{aligned} \quad (1.2.5)$$

再度プロットを試みる。

> plot(f1(x),x=-10..10);



実際に積分してみる。ここでは、鉄則3にしたがって、式を頭から打ち込むのではなく内側からみていくことが肝要である。これは問題を解いていく時に、思考が必ずたどるであろう順番に相当する。

先ず変数に入れた数値をクリアする。

> c:= 'c': g:= 'g': beta:= 'beta':

不定積分でこの関数が積分できることを確認する。

> int(f1(x),x);

$$-\frac{1}{2\beta c e^{\beta c x^2} + \beta g} \left( -\frac{1}{2} \frac{x^3 e^{-\beta c x^2}}{\beta c} + \frac{3}{2} \frac{x e^{-\beta c x^2}}{\beta c} + \frac{1}{4} \frac{\sqrt{\pi} \operatorname{erf}(\sqrt{\beta c} x)}{\beta c \sqrt{\beta c}} \right) \quad (1.2.6)$$

次にx=-alpha..alphaの定積分を実行する。

> int(f1(x),x=-alpha..alpha);

$$-\frac{1}{4} \frac{g \left( 4\alpha^3 e^{-\beta c \alpha^2} \beta c \sqrt{\beta c} + 6\alpha e^{-\beta c \alpha^2} \sqrt{\beta c} - 3\sqrt{\pi} \operatorname{erf}(\sqrt{\beta c} \alpha) \right)}{\beta c^2 \sqrt{\beta c}} \quad (1.2.7)$$

さらにα→∞としてみる。

> limit(int(f1(x),x=-alpha..alpha),alpha=infinity);

$$\lim_{\alpha \rightarrow \infty} \left( -\frac{1}{4} \right) \quad (1.2.8)$$

$$\frac{g \left( 4\alpha^3 e^{-\beta c \alpha^2} \beta c \sqrt{\beta c} + 6\alpha e^{-\beta c \alpha^2} \sqrt{\beta c} - 3\sqrt{\pi} \operatorname{erf}(\sqrt{\beta c} \alpha) \right)}{\beta c^2 \sqrt{\beta c}}$$

ところがこれでは答えを返してくれない。積分した後のそれぞれの項を見るとβ\*c>0を仮定すれば簡単になることが分る。assumeを使って、このような変数の仮定おこなう。

> assume(beta\*c>0);

結果として最初に出した解答を得る。

> limit(int(f1(x),x=-alpha..alpha),alpha=infinity);

$$\frac{3}{4} \frac{\sqrt{\pi} g}{\beta c^2 \sqrt{\beta c}} \quad (1.2.9)$$

## ▼ 式のフォローのデフォルト

Mapleで実際に数式をいじる状況というのは、ほとんどの場合が既知の数式変形のフォローだろう。例えば、論文で「(1)式から(2)式への変形は自明である」とかいう文章で済ましている変形が本当にあっているのかを確かめたい時、一番単純なやり方は自明と言われた前後の式が一致していることを確かめるだけで十分である。

最も単純な確認法は以下の通り、変形の前後の式を手入力してその差をexpandした結果が0か否かでする。

> ex1:=(x-3)^4;

$$\text{ex1} := (x-3)^4 \quad (1.3.1)$$

> ex2:=x^4-12\*x^3+54\*x^2-108\*x+81;

$$\text{ex2} := x^4 - 12x^3 + 54x^2 - 108x + 81 \quad (1.3.2)$$

> expand(ex1-ex2);

$$0 \quad (1.3.3)$$

0ならば式の変形は保証されているので、その導出が間違いでなく誤植などもないことが確認できる。ただ、これだけでは変形の哲学や技法が身に付くわけではない。あくまでも苦し紛れのデフォルトであることは心に留めておくように。