

大学の数学教育における数式処理と $\text{\TeX}$ の活用

Application of computer algebra and $\text{\TeX}$ to the education of university Mathematics

大島利雄

TOSHIO OSHIMA

城西大学 理学部

FACULTY OF SCIENCE, JOSAI UNIVERSITY

1 はじめに

この数年は数学研究の補助や結果の実現のために数式処理 Ris/Asir を活用してきた ([O1, O2]). 2013 年に城西大に移って以降, 大学で数学基礎教育を行う機会が増え, 適切な教材作成のためにコンピュータの活用を考え, 特に 2014 年以降は用いていた数式処理 Risa/Asir によって, 線形代数, 微積分, 離散数学, 初等整数論, 複素関数論, Fourier 解析などの教材を作成した. Risa/Asir においては $\text{\TeX}$ との連携のための関数はいくつか既に作成していたので, それを利用した教材作成を行った. 今回はその中で線形代数で最初に学ぶ行列の基本変形を例にとり, 実際の状況を解説したい (cf. [O3, O4]).

2 行列の基本変形の問題例

行列の「行基本変形」とは, 体 (通常は実数) を成分とする行列に対する操作である. ある行とある行を入れ替える, ある行を (0 でない) スカラー倍する, ある行のスカラー倍を他の行に加える, の 3 つの操作のことで, この操作を続けて行列を簡約な行列に変換することが主目的の操作である. 行の 0 でない成分の先頭は 1 で, この先頭の 1 を含む列の他の成分は全て 0 となっていて, 行番号が増えるに従って行の先頭の 0 の個数が単調増加になっている行列を簡約な行列といい, 行基本変形で得られる簡約な行列は, 行基本変形の手順によらずに出発点の行列からただ一つに定まる.

学生が線形代数を学んでいく上での最初の関門がこの行基本変形で, 連立一次方程式を解くためだけでなく, 逆行列を求めたり, 行列式を計算する上でも基本的な操作である.

城西大学の数学科では, 30 人強のクラスに分かれ, 線形代数を週 1 回の 90 分ずつの講義と演習で 1 年半をかけて学ぶ. 学生の苦手な条件分岐などを含む問題として, 今年度実際に演習で出題した (学生にとって難しめの) 例を一つ挙げよう. この問題は, 演習で出題した後, 完答にしてレポートとして提出する課題を与えた.

問題. 行列 $\begin{pmatrix} k & 3 & a & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & b \end{pmatrix}$ を簡約化せよ. ただし a, b は実数で k は学籍番号の下一桁の数とする.

またこのとき, 連立一次方程式

$$\begin{aligned} kx + 3y + az &= 1 \\ 2x + y &= 1 \\ 3x + 2y + z &= b \end{aligned} \tag{1}$$

を解け.

学生同士が話し合ったり教員に質問可能な状況で演習を行うので、丸写しの答案でなくて自ら考えてもらうため、行列に入れたパラメータが k である。なお、学籍番号下 2 桁の和の一の位、などとすれば、同じ問題に当たる学生の組み合わせが容易に変えられる。

一方、パラメータ k を入れたことで考慮すべき点として、以下が挙げられる。

- k の値によって不適切な問題にならないか？
- k の値の違いによって、問題に難易度の差が生じないか？
- 行基本変形の手順が一意でないだけでなく、問題や最終的な答えも異なることから生じる採点の複雑化。

例として挙げた問題に対する学生の完璧な答案は城西大ではまず期待できないが、答案の採点においては、理解が間違っていて、あるいは不足していて答が違っているのか、また、単なる計算違いなのか、また計算ミスの箇所を見つけたとして、その後の計算や考え方には問題がないのか、など、間違った箇所を個々の答案でチェックして解析し、学生への指導、アドバイスや教育に役立てていく必要がある。

このような作業を手助けするためにコンピュータの手助けを借りている。

3 コンピュータによる行基本変形

行基本変形のプログラムで要請されるのは

- 1 ステップずつ、どのような基本変形を行ったかを示す $\text{\TeX}$ 原稿を出力できる
- コンピュータにとって易しい方法を選ぶのではなく、人が工夫しておこなう基本変形を選ぶ
- 若干ステップ数が増えても、整数成分の行列のときは、なるべく整数を保つ範囲で変形する
- パラメータ（不定元）を含むときは、必要な場合分けを自動的に行う
- パラメータを含む数での割り算はなるべく避ける

などである。

Risa/Asir 上で作成した関数 `mtoupper()` で実現した例を具体的に示すことによって、どのようなことが考慮されているかを見てみる。

先の例 (1) で、 $k = 5$ の場合を得るには

```
[0] M=mat([k,3,a,1],[2,1,0,1],[3,2,1,b])$
[1] os_md.mtoupper(subst(M,k,5),0|step=1,opt=7,dviout=1)$
```

とすると、以下が画面表示される (`dviout=1` の代わりに `dviout=-1` とすると、 $\text{\TeX}$ のソースの出力となる)。

なお、互換性のため、`os_md.` が関数の先頭につけられる。

$$\begin{aligned}
 & \begin{pmatrix} 5 & 3 & a & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & b \end{pmatrix} \xrightarrow{\text{line3} \leftarrow \text{line2}} \begin{pmatrix} 5 & 3 & a & 1 \\ 2 & 1 & 0 & 1 \\ 1 & 1 & 1 & b-1 \end{pmatrix} \xrightarrow{\text{line1} \leftrightarrow \text{line3}} \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 2 & 1 & 0 & 1 \\ 5 & 3 & a & 1 \end{pmatrix} \xrightarrow{\text{line2} \leftarrow \text{line1} \times (2)} \\
 & \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & -1 & -2 & -(2b-3) \\ 5 & 3 & a & 1 \end{pmatrix} \xrightarrow{\text{line3} \leftarrow \text{line1} \times (5)} \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & -1 & -2 & -(2b-3) \\ 0 & -2 & a-5 & -(5b-6) \end{pmatrix} \xrightarrow{\text{line2} \times (-1)} \\
 & \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & 1 & 2 & 2b-3 \\ 0 & -2 & a-5 & -(5b-6) \end{pmatrix} \xrightarrow{\text{line1} \leftarrow \text{line2}} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & -2 & a-5 & -(5b-6) \end{pmatrix} \xrightarrow{\text{line3} \leftarrow \text{line2} \times (2)} \\
 & \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & a-1 & -b \end{pmatrix}
 \end{aligned}$$

If $a = 1$,

$$\begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & -b \end{pmatrix}$$

If $b = 0$,

$$\begin{pmatrix} 1 & 0 & -1 & 2 \\ 0 & 1 & 2 & -3 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

If $b \neq 0$,

$$\begin{aligned} & \xrightarrow{\text{line3} \times = \left(\frac{-1}{b}\right)} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line1} += \text{line3} \times (b-2)} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line2} += \text{line3} \times (-(2b-3))} \\ & \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

If $a - 1 \neq 0$,

$$\begin{aligned} & \xrightarrow{\text{line3} \times = \left(\frac{1}{a-1}\right)} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 1 & \frac{-b}{a-1} \end{pmatrix} \xrightarrow{\text{line1} += \text{line3}} \begin{pmatrix} 1 & 0 & 0 & \frac{-((b-2)a+2)}{a-1} \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 1 & \frac{-b}{a-1} \end{pmatrix} \xrightarrow{\text{line2} -= \text{line3} \times (2)} \\ & \begin{pmatrix} 1 & 0 & 0 & \frac{-((b-2)a+2)}{a-1} \\ 0 & 1 & 0 & \frac{(2b-3)a+3}{a-1} \\ 0 & 0 & 1 & \frac{-b}{a-1} \end{pmatrix} \end{aligned}$$

一方, k をパラメータのままで行基本変形すると

[2] `os_md.mtupper(M,0|step=1,opt=7,dviout=1)$`

$$\begin{aligned} & \begin{pmatrix} k & 3 & a & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & b \end{pmatrix} \xrightarrow{\text{line3} -= \text{line2}} \begin{pmatrix} k & 3 & a & 1 \\ 2 & 1 & 0 & 1 \\ 1 & 1 & 1 & b-1 \end{pmatrix} \xrightarrow{\text{line1} \leftrightarrow \text{line3}} \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 2 & 1 & 0 & 1 \\ k & 3 & a & 1 \end{pmatrix} \xrightarrow{\text{line2} -= \text{line1} \times (2)} \\ & \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & -1 & -2 & -(2b-3) \\ k & 3 & a & 1 \end{pmatrix} \xrightarrow{\text{line3} += \text{line1} \times (-k)} \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & -1 & -2 & -(2b-3) \\ 0 & -(k-3) & a-k & -(kb-k-1) \end{pmatrix} \xrightarrow{\text{line2} \times = (-1)} \\ & \begin{pmatrix} 1 & 1 & 1 & b-1 \\ 0 & 1 & 2 & 2b-3 \\ 0 & -(k-3) & a-k & -(kb-k-1) \end{pmatrix} \xrightarrow{\text{line1} -= \text{line2}} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & -(k-3) & a-k & -(kb-k-1) \end{pmatrix} \\ & \xrightarrow{\text{line3} += \text{line2} \times (k-3)} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & a+k-6 & (k-6)b-2k+10 \end{pmatrix} \end{aligned}$$

If $a = -(k-6)$,

$$\begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & (k-6)b-2k+10 \end{pmatrix}$$

If $k = 6$,

$$\begin{aligned} & \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & -2 \end{pmatrix} \xrightarrow{\text{line3} \times = \left(-\frac{1}{2}\right)} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line1} += \text{line3} \times (b-2)} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ & \xrightarrow{\text{line2} += \text{line3} \times (-(2b-3))} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

If $b = \frac{2(k-5)}{k-6}$,

$$\begin{pmatrix} 1 & 0 & -1 & \frac{-2}{k-6} \\ 0 & 1 & 2 & \frac{k-2}{k-6} \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

If $(k-6)b-2k+10 \neq 0$,

$$\begin{aligned} & \xrightarrow{\text{line3} \times = \left(\frac{1}{(k-6)b-2k+10}\right)} \begin{pmatrix} 1 & 0 & -1 & -(b-2) \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line1} += \text{line3} \times (b-2)} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 2b-3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ & \xrightarrow{\text{line2} += \text{line3} \times (-(2b-3))} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

If $a + k - 6 \neq 0$,

$$\begin{aligned} \text{line3} \times \left(\frac{1}{a+k-6}\right) &\rightarrow \begin{pmatrix} 1 & 0 & -1 & -\frac{(b-2)}{a+k-6} \\ 0 & 1 & 2 & \frac{2b-3}{a+k-6} \\ 0 & 0 & 1 & \frac{(k-6)b-2k+10}{a+k-6} \end{pmatrix} \xrightarrow{\text{line1} += \text{line3}} \begin{pmatrix} 1 & 0 & 0 & \frac{-((b-2)a+2)}{a+k-6} \\ 0 & 1 & 2 & \frac{2b-3}{a+k-6} \\ 0 & 0 & 1 & \frac{(k-6)b-2k+10}{a+k-6} \end{pmatrix} \\ \text{line2} -= \text{line3} \times (2) &\rightarrow \begin{pmatrix} 1 & 0 & 0 & \frac{-((b-2)a+2)}{a+k-6} \\ 0 & 1 & 0 & \frac{(2b-3)a+k-2}{a+k-6} \\ 0 & 0 & 1 & \frac{(k-6)b-2k+10}{a+k-6} \end{pmatrix} \end{aligned}$$

となり、 $k = 6$ の場合は若干易しくなることが分かる（期末試験などではないので、若干の難易度の差があった方が変化があって面白いであろうと判断した）。

- 基本変形を $\text{line2} -= \text{line1} \times (2)$ のように C での演算式の形で表示するようにしているのは、演算でどの行を変更するかを明示するため（ここで混乱する学生が多いのでこのような表示にした。なお、よく用いられる ①+② といった表示では、混乱を生みやすい）。
- `mtoupper()` の第 2 引数に行数 n の符号を変えた $-n$ を指定すると、右に n 次正方単位行列を付加し、（その部分を組み入れない）基本変形を行う（正の整数 m を指定すると、最後の m 列を組み入れない行基本変形を行う）。
これにより変換行列が求められる。特に元が正則行列ならば、行基本変形を用いて逆行列が得られる。
- `opt` を指定しないと、行のスカラー倍は行わず、行の交換では下方に移動した行は -1 倍して（行列式の計算に有効）上三角型へ変形する。
`opt=1` では、さらに行の先頭が 0 でない列の他の行の成分を 0 にする。
`opt=2` では、さらに行の先頭の 0 でない成分を、行のスカラー倍で 1 にする。
`opt=3` では、さらに次節優先手順の 2, 3 のみを考慮。
`opt=4` では、さらに整数成分に対し、なるべく分数が現れないように変形。
`opt` の値に 5 以上を設定することにより、パラメータに依存した場合分けに対応する。値が大きくなるとより細かな配慮を行う（値が 6 以上では、2 つのパラメータが同事に絡む場合分けに対応）。
- `step=1` で、各ステップ毎の変形過程を 1 ステップ毎に表示する。
- オプション `tab=k` を指定すると、パラメータの値による場合分けが生じたときの各段階でのインデントの幅を k mm とする。
- オプション `pages=1` を指定することにより、複数ページに渡る原稿を許す。
- オプション `cr=` を適当に指定することにより、基本変形が 1 ステップずつ表示されるようなスライド原稿を作成することができる。講義で基本変形を説明する際に用いるのが目的。

4 行基本変形の優先手順

`mtoupper()` が、どのような優先順位で各基本変形を選んでいくかを以下に述べる。

実際に行っていることを正確に述べるのは複雑過ぎるので（特にパラメータが入って、条件分岐が要請される場合）、以下はその概略である。

1. 行列 $M = (m_{j,l})$ は $k-1$ 列目まで基本変形が終了した行列で、その j_0-1 行目までの基本変形が完了しているとする。このとき $k-1$ 列までの成分は j_0 行目以降は 0 となっていて、 j_0-1 行目までの行の 0 でない先頭成分は 1 で、それは $k-1$ 列目以下で、その列の他の成分は 0 になっている。また k 列目には j_0 行目またはそれ以降に初めて 0 でない成分 $m_{j,k}$ があるとする。

以下のように j_0 行目またはそれ以降の 0 でない t 行目の成分 $m_{t,k}$ を調べて基準の j_1 行目を決め、(必要なら) 行交換してそれを j_0 行目に移動する。得られた行列 $M = (m_{j,\ell})$ において $m_{j_0,k}$ で j_0 行目を割り、 j_0 行目のスカラー倍を他の行に加えて少なくとも k 列目までの基本変形を終了させる。

2. j 行目またはそれ以降の成分 $m_{t,k}$ で 1 となるものがあればその最初の t を j_1 行目とおく。
3. そうでなくて j 行目またはそれ以降の成分 $m_{t,k}$ で -1 となるものがあればその最初を j_1 行目とおく。
4. そうでなくて j 行目またはそれ以降の成分 $m_{t,k}$ が正整数で、その行を $m_{t,k}$ で割ったものが整数行ベクトルになるものがあればその最初の行を j_1 行目とする。
5. そうでなくて j 行目またはそれ以降の成分 $m_{t,k}$ が負整数で、その行を $m_{t,k}$ で割ったものが整数行ベクトルになるものがあればその最初の行を j_1 行目とする。
6. そうでなくて j 行目またはそれ以降の成分 $m_{t,k}$ が 0 でなくて、その行を $m_{t,k}$ で割ったものが整数行ベクトルになるものがあればその最初の行を j_1 行目とする。
7. そうでなければ、 t 行目の成分 $m_{t,k}$ が整数で、 k 行目以降の t 行目と異なる j 行目に t 行目の整数倍を足して (j,k) 成分を 0 に出来るか調べ、可能ならそれを行う。
8. そうでなければ、 t 行目の成分 $m_{t,k}$ が整数で、 j 行目以降のある行を t 行目に足すか引くかして (t,k) 成分を 1 に出来るか調べ、可能ならそれを行って $j_1 = t$ とする。ただし、 j 行目と t 行目に変数を含まないものを優先する。
9. そうでなければ、 t 行目の成分 $m_{t,k}$ が整数で、 j 行目以降のある行目に j 行目以降のある行の整数倍を足すか引くかして (t,k) 成分を 1 に出来るか調べ、可能ならそれを行ってその行を $j_1 = t$ とする。ただし、 j 行目と t 行目に変数を含まないものを優先する。
10. そうでなければ、 t 行目の成分 $m_{t,k}$ が整数で、 j 行目以降のある行に j 行目以降のある行を足すか引くかして (t,k) 成分を -1 に出来るか調べ、可能ならそれを行って $j_1 = t$ とする。ただし、 j 行目と t 行目に変数を含まないものを優先する。
11. そうでなければ、 t 行目に j 行目以降のある行の整数倍を足すか引くかして (t,k) 成分を -1 に出来るか調べ、可能ならそれを行って $j_1 = t$ とする。ただし、 j 行目と t 行目に変数を含まないものを優先。
12. そうでなければ最初に現れた整数成分 $m_{t,k}$ の行を、整数の行がなければパラメータを含まない数 $m_{t,k}$ の最初の行を j_1 行とする。
13. `opt>4` のとき、 k 列目の有理式 $m_{t,k}$ の分子がある一つパラメータの多項式となっているとき、行変形によって k 列目の成分の分子の多項式の最低次数を下げる事ができればそれを行うことを続ける。
14. `opt>4` のとき、 k 列目の成分で分子が数となる $m_{t,k}$ があれば、その最初の行 t によって $j_1 = t$ とする。
15. `opt>4` のとき、有理式 $m_{t,k}$ で分子の最大公約元の多項式の根が有理数で与えられるときは、その項が 0 となる場合をまず扱って行変形の最終形まで求める。そのあと $j_1 = t$ とする。
16. `opt>4` のとき、 $m_{t,k}$ が 0 になるための条件が、ある変数とその変数を含まない変数の多項式や分母が 0 でない有理式 (ただし `opt=5` のときは有理数) で与えられるものがあるかどうか調べ、そのような t が存在すれば 0 になる場合を最終形まで求め、次に 0 にならない場合を調べるため $j_1 = t$ とする。
17. 上に該当しなければ、成分 $m_{t,k}$ の型が最小となるものが最初に現れる行 t に対して $j_1 = t$ とする。

[3] `os_md.mtupper(os_md.s2m("32,53"),-2|step=1,opt=7,dviout=1)$`

$$\begin{pmatrix} 3 & 2 & 1 & 0 \\ 5 & 3 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line2} \leftarrow \text{line1} \times (2)} \begin{pmatrix} 3 & 2 & 1 & 0 \\ -1 & -1 & -2 & 1 \end{pmatrix} \xrightarrow{\text{line1} \leftrightarrow \text{line2}} \begin{pmatrix} -1 & -1 & -2 & 1 \\ 3 & 2 & 1 & 0 \end{pmatrix} \xrightarrow{\text{line1} \times (-1)} \begin{pmatrix} 1 & 1 & 2 & -1 \\ 3 & 2 & 1 & 0 \end{pmatrix} \\
 \xrightarrow{\text{line2} \leftarrow \text{line1} \times (3)} \begin{pmatrix} 1 & 1 & 2 & -1 \\ 0 & -1 & -5 & 3 \end{pmatrix} \xrightarrow{\text{line2} \times (-1)} \begin{pmatrix} 1 & 1 & 2 & -1 \\ 0 & 1 & 5 & -3 \end{pmatrix} \xrightarrow{\text{line1} \leftarrow \text{line1} - \text{line2}} \begin{pmatrix} 1 & 0 & -3 & 2 \\ 0 & 1 & 5 & -3 \end{pmatrix}$$

整数成分の行列の場合、上のように整数の 1 回の割り算で ± 1 が得られる場合は、それを行うが、それ以上の回数の割り算は行わない（ユークリッドの互除法まで行うのはやり過ぎと判断した）。

[4] `os_md.mtouppeer(os_md.s2m("52,73"),-2|step=1,opt=7,dviout=1)$`

$$\begin{pmatrix} 5 & 2 & 1 & 0 \\ 7 & 3 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line1} \times = (\frac{1}{5})} \begin{pmatrix} 1 & \frac{2}{5} & \frac{1}{5} & 0 \\ 7 & 3 & 0 & 1 \end{pmatrix} \xrightarrow{\text{line2} -= \text{line1} \times (7)} \begin{pmatrix} 1 & \frac{2}{5} & \frac{1}{5} & 0 \\ 0 & \frac{1}{5} & -\frac{7}{5} & 1 \end{pmatrix} \xrightarrow{\text{line2} \times = (5)} \begin{pmatrix} 1 & \frac{2}{5} & \frac{1}{5} & 0 \\ 0 & 1 & -7 & 5 \end{pmatrix} \xrightarrow{\text{line1} -= \text{line2} \times (\frac{2}{5})} \begin{pmatrix} 1 & 0 & 3 & -2 \\ 0 & 1 & -7 & 5 \end{pmatrix}$$

パラメータを含んだ場合分けは、以下のように体の拡大を必要としない場合分けまで。

[5] `os_md.mtouppeer(mat([a^4+2*a^2*b^2]),0|step=1,opt=7,dviout=1)$`

$$\begin{aligned} & \begin{pmatrix} a^2(a^2 + 2b^2) & b \\ 0 & 1 \end{pmatrix} \\ \text{If } a = 0, & \\ & \begin{pmatrix} 0 & b \\ 0 & 0 \end{pmatrix} \\ \text{If } b = 0, & \\ & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \\ \text{If } b \neq 0, & \\ & \xrightarrow{\text{line1} \times = (\frac{1}{b})} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \\ \text{Assume } a(a^2 + 2b^2) \neq 0, & \\ & \xrightarrow{\text{line1} \times = (\frac{1}{a^2(a^2+2b^2)})} \begin{pmatrix} 1 & \frac{b}{a^2(a^2+2b^2)} \\ 0 & 0 \end{pmatrix} \end{aligned}$$

5 採点の手助け

`mtouppeer()` が行基本変形で簡約な行列への変換のよい手順を与えたにしても、実際の学生の答案における手順とは異なることがほとんどであろう。そのような答案における学生の計算ミスの箇所のチェックのために作成した Risa/Asir の関数が `transm()` であって、行列の変換をインタラクティブに行うことができる。

- `dviout=1` を指定すると、 $\text{T}_{\text{E}}\text{X}$ を用いて結果のインタラクティブで綺麗な画面表示ができる。

表示される？ の後に 1 行キー入力して指示することによって基本変形する。例えば

空 : コマンドの表示

2,5 : 2 行目と 5 行目の入れ替え

2,5,-2 : 2 行目に 5 行目の -2 倍を加える

2,2,-2 : 2 行目を -2 倍する

2,5,0 : 2 行目に 5 行目のスカラー倍を加えて、2 行目における 5 行目の先頭の 0 でない列を 0 に

r,2,5 : 2 列目と 5 列目の入れ替え（列変形を行うには先頭に r をつける）

s,x,2 : x に 2 を代入する

t : 転置する

0 : 最初の行列に戻る

f : 一つ前の行列に戻る

g : 次の行列へ（f の直後のみ可能）

a : 以降、自動的に行基本変形を行う

- A : 上と同じだが dviout=1 のときは, $\text{\TeX}$ を用いた画面表示となる
 q : 終わり (変形の過程の行列をリストにして返す)

実際の Risa/Asir での実行例は

```
[6] L=os_md.transm(mat([2,3,1,-1,a],[3,2,2,1,1],[4,0,3,4,1],[5,4,4,1,1]))$
[ 2 3 1 -1 a ]
[ 3 2 2 1 1 ]
[ 4 0 3 4 1 ]
[ 5 4 4 1 1 ]
?                               /* 空入力でコマンドの説明が出る */
2,5 : line2 <-> line5
2,5,-2 ; line2 += (-2)*line5
2,2,-2 : line2 *= -2
2,5,0 : line2 += (?)*line5 for reduction
....
? 3,2,-1                       /* 2行目の-1 倍を 3 行目に加える */
[ 2 3 1 -1 a ]
[ 3 2 2 1 1 ]
[ 1 -2 1 3 0 ]
[ 5 4 4 1 1 ]
? 1,3                           /* 1 行目と 3 行目の入れ替え */
[ 1 -2 1 3 0 ]
[ 3 2 2 1 1 ]
[ 2 3 1 -1 a ]
[ 5 4 4 1 1 ]
? 2,1,0                         /* 1 行目によって 2 行目を基本変形する */
[ 1 -2 1 3 0 ]
[ 0 8 -1 -8 1 ]
[ 2 3 1 -1 a ]
[ 5 4 4 1 1 ]
? f                             /* 1 ステップ戻る */
[ 1 -2 1 3 0 ]
[ 3 2 2 1 1 ]
[ 2 3 1 -1 a ]
[ 5 4 4 1 1 ]
? s,a,1                         /* a に 1 を代入 */
[ 2 3 1 -1 1 ]
[ 3 2 2 1 1 ]
[ 4 0 3 4 1 ]
[ 5 4 4 1 1 ]
? q                             /* 終わり */
```

注意. このほか問題の自動生成も考えられるが, その程度は教員の工夫力向上のために残しておくのがよいかもしれない.

6 行列の入力

学生に与える問題では, 成分が小さな整数の行列であることが多い. そのような行列を容易に入力する関数 `s2m()` を作成した. 例で示そう.

$$\begin{pmatrix} 2 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad : \text{s2m}("21,02,0^2-1")$$

文字列で上の行列を入力するこの例では、行を「,」で区切って入力する。行の長さは入力した最大のものとなり、後部の0は省略可能。数 n が m 個続くときは n^m と書いてよい。2桁の数は(12)のように区切って入力する。また、 $a, b, \dots$ は、10, 11, ... と解釈される。さらに、 $2/b$ は $\frac{2}{11}$ と解釈される。

文字列でなくて `mat()` と似た形での入力、ただし行の後部の0を省略して `s2m([[2,1],[0,2],[0,0,-1]])` としてもこの行列が入力される。

以下のように `Risa/Asir` が出力したような文字列形式の行列（コピー・ペースト可）での入力もできる。

$$\begin{pmatrix} x^2 & 3 \\ z & w+1 \end{pmatrix} \quad : \text{s2m}("[x*x \ 1+2] \ [z \ w+1]")$$

成分が a_{ij} となるような一般行列も配布資料で使うことが多い。このような行列の入力も容易に行うのが `mgen()` である（対角行列のときは `diagm()` がより容易）。

対角行列

$$\begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \quad : \text{diagm}(3, [a,b,c]) \quad (\text{対角成分を与える})$$

$$\begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix} \quad : \text{diagm}(3, [\text{lambda}]) \quad (\text{スカラー行列})$$

$$\begin{pmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{pmatrix} \quad : \text{diagm}(3, a) \quad (\text{対角成分の自動添え字})$$

一般行列

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \quad : \text{mgen}(3,3,a,1) \quad (\text{二重添え字の一般行列})$$

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix} \quad : \text{mgen}(3,3,a,0) \quad (\text{二重添え字の一般行列})$$

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \quad : \text{mgen}(3,3,[a,b,c],1) \quad (\text{一重添え字の一般行列})$$

$$\begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix} \quad : \text{mtanspose}(\text{mgen}(3,3,[a,b,c],1)) \quad (\text{一重添え字の一般行列})$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{pmatrix} \quad : \text{mgen}(3, "symmetric", a, 1) \quad (\text{二重添え字の一般対称行列})$$

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ a_2 & b_2 & b_3 \\ a_3 & b_3 & c_3 \end{pmatrix} \quad : \text{mgen}(3, "symmetric", [a,b,c], 1) \quad (\text{一重添え字の一般対称行列})$$

$$\begin{pmatrix} 0 & a_{12} & a_{13} \\ -a_{12} & 0 & a_{23} \\ -a_{13} & -a_{23} & 0 \end{pmatrix} \quad : \text{mgen}(3, "skew", a, 1) \quad (\text{二重添え字の一般歪対称行列})$$

$$\begin{pmatrix} 0 & a & 0 \\ 0 & 0 & a \\ 0 & 0 & 0 \end{pmatrix} \quad : \text{mgen}(3, "highdiag", [a], 1) \quad (\text{対角成分の一つ上})$$

$$\begin{pmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{pmatrix} : \text{diagm}(3, [\text{lambda}]) + \text{mgen}(3, \text{"highdiag"}, [1], 1)$$

7 他の行列変形のプログラム

行列の基本変形で `mtoupper()` のように計算の各ステップについても $\text{T}_{\text{E}}\text{X}$ を使った表示結果が得られるものに、ユークリッド環成分の行列の行と列の基本変形を行うために作成した `mdivisor()` がある。

行列のジョルダン標準形を求める際や、有限生成アーベル群の基本定理での具体的計算に用いられる。`mdivisor()` が対応しているユークリッド環は以下の3つである。

- 整数の環
- 多項式環で、係数是有理数や他の変数の有理関数体
- 線型常微分作用素の環（係数は変数やパラメータの有理関数体）

上の3つめは非可換環で、決定系の場合は *cyclic vector* の存在定理（アーベル群のときは、有限アーベル群の基本定理に対応）として知られている。ここでの場合は、左右からの基本変形の繰り返し（可逆行列の掛算）で対角成分が一つを除いて全て1か0の行列に変形される（cf. [O1, Lemma 1.10]）

右側と左下に単位行列を加えたステップ単位での変形の $\text{T}_{\text{E}}\text{X}$ を使った表示もできる。

途中のステップを省いた最終結果の例を挙げる。

```
[7] os_md.mdivisor(mat([dx,0,0],[0,dx,0],[0,0,dx]),[x,dx]|dviout=3)$
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \partial^3 \end{pmatrix} = P \begin{pmatrix} \partial & 0 & 0 \\ 0 & \partial & 0 \\ 0 & 0 & \partial \end{pmatrix} Q,$$

$$P = \begin{pmatrix} -x & 1 & 0 \\ -\frac{1}{2}x\partial - 1 & \frac{1}{2}\partial & -\frac{1}{2}x^2\partial - 2x \\ -x\partial^2 - 3\partial & \partial^2 & -x^2\partial^2 - 6x\partial - 6 \end{pmatrix} = \begin{pmatrix} \partial & -x^2\partial^2 - 4x\partial - 2 & \frac{1}{2}x^2\partial + 2x \\ x\partial + 1 & -x^3\partial^2 - 4x^2\partial - 2x & \frac{1}{2}x^3\partial + 2x^2 \\ 0 & \partial & -\frac{1}{2} \end{pmatrix}^{-1},$$

$$Q = \begin{pmatrix} 1 & -x^2\partial - 2x & \frac{1}{2}x^2\partial^3 + x\partial^2 - \partial \\ x & -x^3\partial - x^2 & \frac{1}{2}x^3\partial^3 + \frac{1}{2}x^2\partial^2 - x\partial + 1 \\ 0 & 1 & -\frac{1}{2}\partial^2 \end{pmatrix} = \begin{pmatrix} -x\partial & \partial & 0 \\ -\frac{1}{2}x\partial^2 - \partial & \frac{1}{2}\partial^2 & -\frac{1}{2}x^2\partial^2 - 2x\partial \\ -x & 1 & -x^2 \end{pmatrix}^{-1}.$$

上で dx や ∂ は $\frac{d}{dx}$ を表す。引数の2番目の $[x, \text{dx}]$ は、 $\text{dx} \cdot x - x \cdot \text{dx} = 1$ を、オプションの `dviout=3` は結果のみの $\text{T}_{\text{E}}\text{X}$ による表示を意味する。

上の結果は、以下の2つの方程式の同型対応を具体的に与えている。

$$\frac{du_1}{dx} = \frac{du_2}{dx} = \frac{du_3}{dx} = 0 \Leftrightarrow \frac{d^3v}{dx^3} = 0.$$

8 成績処理

私は学生の毎回の演習の成績や試験の結果をエクセルで管理している。エクセルと *Risa/Asir* のデータのやりとりを行うため、関数 `readcsv()` と `tocsv()` を作成し、CSV形式でデータ入出力を行っている。

`readcsv()` では、CSV形式データを読み込むが、表のどの列をどのような形式で読み込むか、あるいは読み込まないかが指定できる。またデータの区切り記号なども（汎用性のため）指定可能である。ShiftJISなどの文字コードにも対応している。

```
[0] M=[74,71,47,80,66],[54,71,50,76,45],[75,70,78,x,69],[35,44,74,35,37],
[59,x,85,75,52],[92,55,70,61,45],[70,79,77,76,55]]$
[1] S=os_md.distpoint(M|opt="average",title="線形代数（中間試験）");
\begin{tabular}{|c|c|c|c|c|c|}
\multicolumn{6}{c}{線形代数（中間試験）}\hline
平均点& 標準偏差 & 最低点 & 最高点 & 受験人数 & 欠席者\\
$63.7$& $15.1$& $35$& $92$& $33$& $2$\\
\end{tabular}
[2] os_md.dviout(S)$
```

上の [0] では点数のリストを M に直接代入しているが（実際は短いので記録しておいたものからのコピー・ペーストで入れることが多い。内部の区切りの [] は分けて見やすくするためで、無視される）。

エクセルなどに点数のデータが入っているときにそれを読み込むには、例えば *fname* というファイルに CSV 形式で出力し、そのデータの *n* 項目が点数のデータとすると

```
M=os_md.readcsv(fname|col=n,eval=n)
```

によって点数のデータが M にリストとして読み込まれる。

[1] の最後が ; となっているので、実行結果、すなわち作成された T_EX のソース S が表示される。\$ とすると結果は表示されない。

データに非負の数でないものがあれば、それは試験の欠席者のデータとみなす。

[2] の T_EX による S の表示（文書）は以下ようになる。

線形代数（中間試験）

平均点	標準偏差	最低点	最高点	受験人数	欠席者
63.7	15.1	35	92	33	2

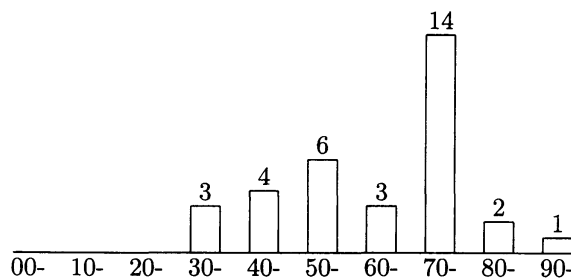
```
[3] S=os_md.distpoint(M|title="点数分布")$
[4] os_md.dviout(S)$
```

点数分布

00-09	10-19	20-29	30-39	40-49	50-59	60-69	70-79	80-89	90-100
0	0	0	3	4	6	3	14	2	1

点数の分布の T_EX によるグラフ表示には、オプションに `opt="graph"` を指定する。

```
[5] S=os_md.distpoint(M|opt="graph")$
[6] os_md.dviout(S)$
```



- 点数分布は 10 点刻みがデフォルトであるが、5 点刻みとするにはオプション `div=5` を指定すればよい。
- 点数分布の棒グラフのサイズを `size=[w,h,r,s]` で指定できる。 *w* は横幅、 *h* は最も高い棒グラフの高さ、 *r* は棒幅と隣との空きとの比率、 *s* は文字列との空きの高さで、最初の 2 項以外は省略できる。

グラフ表示には $\text{T}_\text{E}\text{X}$ のパッケージの Xy-pic または TikZ を用いるが、長さの単位は前者は mm を後者は cm とする。

9 再び行列の変形

先の節の例のように、表を $\text{T}_\text{E}\text{X}$ のタブ環境として出力する機能を `os_muldif.rr` は持っている。表は Risa/Asir のリスト（のリスト）や行列として扱われ、その成分は数値や文字列など任意のものが許される。

例えば、 $\sin$ と $\tan$ の数値表を 1° 刻みで作成することを考えてみる。表は、度の値と三角関数の値で、91 行 3 列の行列とみなせる。そのままの表では縦長過ぎるので折り返して 2 段や 3 段の、6 列や 9 列の表にするのが適当であろう。また 1 行目に「度」とか $\sin$ など説明の行を付加するのが適当であろう。

縦長の表を分割して横に繋げる、横長の表を折り返して縦に繋げる、表から行や列を抜き出したり入れ替えたりして新しい表を作る、2 つ以上の表を結合したり転置した表にする、などは表を行列とみなすと、行列の変形と考えてよい。

`os_muldif.rr` には、このようなことを実現する関数 `ltotex()`、`madjust()`、`mperm()`、`newbmat()`、`mtranspose()` が備わっている（詳しくは `os_muldif.pdf` を参照）。

今回は、行列の基本変形に関連して、`os_muldif.rr` に含まれる関数 `mtoupper()`、`transm()`、`s2m()`、`diagm()`、`mgen()`、`mdivisor()` の他、成績処理に関連して `readcsv()` と `distpoint()` を紹介した。現在 `os_muldif.rr` には、326 個余りの関数があり、そのマニュアル `os_muldif.pdf` と共に公開している ([O4])。

参考文献

- [O1] T. Oshima, *Fractional calculus of Weyl algebra and Fuchsian differential equations*, MSJ Memoirs **28**, Mathematical Society of Japan, Tokyo, 2012.
- [O2] 大島利雄, 数式処理による数学研究とプレゼンテーション, 数式処理とその周辺分野の研究, 数理解析研究所講究録 **1907** (2014), 97–109.
- [O3] 大島利雄, Risa/Asir による曲線と関数グラフの描画, 数式処理とその周辺分野の研究, 数理解析研究所講究録 **1955** (2015), 102–113.
- [O4] 大島利雄, `os_muldif.rr` および `os_muldif.pdf`, a library for computer algebra Risa/Asir , 2007–2015, <ftp://akagi.ms.u-tokyo.ac.jp/pub/math/muldif/>