

MaximaOnLineの活用術

— 独力で数学学習を進めるために —

北 秀和

0. はじめに

ICTを活用して、授業前学習を進めるための教材として、MaximaOnLineを活用したスクリプト例を、高等学校の教科書の例題等について作成してきた。取り敢えず作成したものを、数学科教育法a,b,c,dの課題として位置づけ、受講者の取組みの様子を踏まえて、逐次改善を進めてきた。その現段階のスタイルをまとめておきたい。

1. 定数・変数、コマンドなど

1) 以下の例を、それなりにイメージできるものとする。

- ①定数（数値、文字） 3, 3.14, %pi, %e, “Hello!”, ...
- ②変数 x, y, hensuu, Hensuu1, ...
- ③代入コマンド 「:」
- ④コマンド print(), sin(), ...
- ⑤演算子 「*」, 「+」,
- ⑥実行子 「;」 「\$」

例(以下、出力結果の手前までを、maxima.cesga.esの黄色部分にCopy&Pasteし、Clickしてみること)

```
x:%pi; x:%pi$
```

出力結果

```
(%i1) x:%pi;
```

```
(%o1)  $\pi$ 
```

```
(%i2) x:%pi$
```

コメント

「;」 実行後の戻値を表示

「\$」 実行後の戻値を表示しない

2) 文 (単文、複文)

① 文 実行子で実行される一まとまり

② 単文 代入コマンドが1以下で、代入コマンドがあれば、最後にそれが実行されるもの

例

```
x:%pi;  
print(x)$
```

出力結果

```
(%i1) x:%pi;
```

```
(%o1)  $\pi$ 
```

```
(%i2) print(x)$
```

```
 $\pi$ 
```

コメント

printコマンドは、入出力番号なしで、斜体で出力

③ 複文 最後ではないところで実行される代入コマンドを含み、「,」を実行子に変更することなどでは二文に分割できないもの

例

```
print(sin((x:%pi)/3) );
```

出力結果

```
(%i1) print(sin((x:%pi)/3) );
```

```

$$\frac{\sin(\frac{\pi}{3})}{2}$$

```

```
(%o1)  $\frac{\sqrt{3}}{2}$ 
```

コメント

xに「%pi」という値が代入されて、それを3で割ったものの正弦の値が斜体で小さく表示
正弦は弧度法で計算される。

④ ブロック 二文以上からなり、全体を()で括って、()内の実行子を「,」に変え、終わりの「)」の直後にある実行子により実行されるもの

例

```
(x:%pi, y:sin(x/3), print(y) );
```

```
(x:%pi, y:sin(x/3), print(y) )$
```

出力結果

```
(%i1) (x:%pi, y:sin(x/3), print(y) );
```

$$\frac{\sqrt{3}}{2}$$

```
(%o1)  $\frac{\sqrt{3}}{2}$ 
```

```
(%i2) (x:%pi, y:sin(x/3), print(y) )$
```

$$\frac{\sqrt{3}}{2}$$

コメント

前例と同じ処理をブロックで表現したもの

ブロック内の各文の戻値は、実行子が「;」でも非表示で、最後のコマンドの戻値だけが、戻り値としてスクリプト内部に保持され、「;」の場合だけ出力番号付きで表示される

2. コメントアウト

1) コメントアウト「/*...*/」

① コメントアウトと実行子

例

```
x:%pi/3$ print(sin(x) )$ /*コメントアウトされる*/
```

```
x:%pi/3$ print(sin(x) ) /*コメントアウトされる*/$
```

出力結果

```
(%i1) x:%pi/3$
```

```
(%i2) print(sin(x) )$
```

$$\frac{\sqrt{3}}{2}$$

```
(%i3) /*コメントアウトされる*/  
x:%pi/3$
```

```
(%i4) print(sin(x) ) /*コメントアウトされる*/$
```

$$\frac{\sqrt{3}}{2}$$

コメント

実行子の出現ごとにコマンド実行

例

```
/*コメントアウト単独実行不可*/$
```

出力結果

Programming error detected. Check your input.

コメント

コメントアウトを単独で実行できない。

② コメントアウトと行

例

```
x:%pi/3$ /* print(sin(x) )$
```

```
x:%pi/3$ */ print(sin(x) )$
```

出力結果

```
(%i1) x:%pi/3$
```

```
(%i2) /* print(sin(x) )$  
      x:%pi/3$ */ print(sin(x) )$
```

```
      sqrt(3)  
      -----  
      2
```

コメント

行を超えてコメントアウトできる。

3. リスト

1) リストとprintコマンド、ブロック

① リスト

定数、変数、戻値などを、ひとまとまりのセットにしたもの

例

```
L:[x, x2:%pi/3, sin(x2), "Hello!"]$  
print(L)$
```

出力結果

```
(%i1) L:[x, x2:%pi/3, sin(x2), "Hello!"]$
```

```
(%i2) print(L)$
```

```
      %pi  sqrt(3)  
[x, ---, ----, Hello!]  
      3      2
```

コメント

Lには、

数値代入されていない変数xの項は、xのままを、
変数x2に数値代入した項は、代入の戻値を、
数値代入された変数の関数の項は、関数の戻値を、
文字定数の項は、文字定数のままを
それぞれ順に配置したリストが、代入される。

② printコマンド

printコマンドは、実行子「;」「\$」の区別に関わらず、小さな斜体で出力される。
実行子が「;」の場合、printコマンドの戻値が通常出力される。

例

```
ev11:["ev11","Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",  
/**/float([KKK,subst([a=%pi/4, b=%pi/6], [  
"Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ]$  
print(ev11)$
```

出力結果

```
(%i1) ev11: ["ev11", "Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",  
            /**/float([KKK,subst([a=%pi/4, b=%pi/6], [  
            "Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ]$
```

```
(%i2) print(ev11)$
```

```
ev11, Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b),  
[KKK, [Ge=, 0.96592582628907, =, 0.96592582628907, (加法定理)]]]
```

コメント

最初の文で、リストが作成され、最後にev11に代入されているので、

最初の文は、単文

リストが3重になっており、

最も外側のリストの

第1項は、リストの名前"ev11"を、

第2項は、与式Geの展開を文として記述したもの、

第3項は、外側2番目のリストで、

その第1項は、数値確認のリストである

ことを示す記号KKKがあり、

第2項は、最も内側のリストで、

その第1項は、与式"Ge="

その第2項は、a,bに指定の数値を代入したsin(a+b)の値

その第3項は、"="

その第4項は、a,bに指定の数値を代入した

sin(a)*cos(b)+cos(a)*sin(b)の値

その第5項は、根拠となる定理の名前"加法定理"

を並べたものである

外側2番目のリストの各項は、実数（小数）表記となっている。

例

```
print("ev11","Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",  
/**/float([KKK,subst([a=%pi/4, b=%pi/6], [  
            "Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ]$
```

```
"Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ) $
```

出力結果

```
(%i1) print("ev11", "Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",  
  /**/float([KKK,subst([a=%pi/4, b=%pi/6], [  
    "Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ) $  
  
ev11 Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)  
      [KKK, [Ge=, 0.96592582628907, =, 0.96592582628907, (加法定理)]]
```

コメント

printコマンドで

①"ev11",②"Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",

③/**/float([KKK,subst([a=%pi/4, b=%pi/6], [
 "Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)"])))

の3つを出力している。1つ目、2つ目、3つ目の間に「,」がないことに注意

出力された3つ目は、リスト。リストの出力では、成分の間に「,」がある。

リスト出力であるか否かに関わらず、「」は消されて出力されている。

print文は、代入コマンドなしに実行されるので、単文

なお、例の末尾の「]))) 」 (「_」は半角空白を示す。) は、

対応する左括弧が、外側から順に、1番目と2番目は連続せず、2番目と3番目は連続し、3

番目・4番目・5番目は連続せずに、5番目が一番内側の括弧であること

を示している。

このルールにしたがって記述すると、カッコの対応の不整合に気づきやすくなる。

③ ブロック

リスト代入文とそのprint文をブロックにしておくと、出力が一括されるので、処理の流れが見やすくなる。

例

```
(SHORI1:"加法定理の展開",  
ev11:["ev11", "Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",  
  /**/float([KKK,subst([a=%pi/4, b=%pi/6], [  
    "Ge=",sin(a+b),"=",sin(a)*cos(b)+cos(a)*sin(b),"(加法定理)" ] ) ) ]),  
print(ev11),  
/**/End:"Shori1" )$
```

出力結果

```
(%i1) (SHORI1:"加法定理の展開",
      ev11:["ev11","Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b)",
      /**/float([KKK,subst([a=%pi/4, b=%pi/6], [
      "Ge=",sin(a+b), "=",sin(a)*cos(b)+cos(a)*sin(b), "(加法定理)" ] ) ]),
      print(ev11),
      /**/End:"Shori1" )$

[ev11, Ge=sin(a+b)=sin(a)*cos(b)+cos(a)*sin(b),
 [KKK, [Ge=, 0.96592582628907, =, 0.96592582628907, (加法定理)]]]
```

コメント

前出の例をブロックにしたもの

2つの単文を「(」 「)」で括り、

各単文の実行子を「,」に変更

最後に、実行子「\$」で実行している。

4. ブロックによる分節化

1) 分節化したスクリプト

自力学習を進めるためのMaximaアプリの最終形は、ブロックにより分節化したスクリプトとなっている。

その最も簡潔なスタイルは、以下の例のように分節化したものである。

この6分節のスタイルを例として、説明する。

```
/* 加法定理の展開 */
/*(kill(all), print("インストールMaximaでは、行頭の/*を、次行頭にCut
& Paste"))$ */
/*2*/
(/=* / MONDAI:"sin(a+30)を加法定理で展開すると？(度数法)",
print(MONDAI),
/**/End:"Mondai")$
/*3*/
(KANSUU:"",
sind(t):=sin(t/180*Pi),cosd(t):=cos(t/180*Pi),
/**/End:"Kansuu")$
/*4*/
(TEISUU:"",
print([Pi:%pi,deg:25/180*Pi]),
/**/End:"Teisuu")$
/*5*/
((/*=* / TEJUN1: "1 解析的 sin(a+30)を加法定理で展開すると？",
subl:[a=deg],
```



```

print(["subl",subl]) ),
(ev11:["ev11","Ge: sin(a+30)=sin(a)*cos(30)+cos(a)*sin(30)",
/**/float([KKK,subst(subl, [
"Ge: ",sind(a+30), "=",sind(a)*cosd(30)+cosd(a)*sind(30),"(加法定理)" ] ) ] ),
print(ev11) ),
(ev12:["ev12"," =sin(a)*sqrt(3)/2+cos(a)*1/2 =(sqrt(3)*sin(a)+cos(a))/2",
/**/float([KKK,subst(subl, [
" =",sind(a)*sqrt(3)/2+cosd(a)*1/2 , "=",(sqrt(3)*sind(a)+cosd(a))/2,"(30°の正弦・余弦)" ] ) ] ),
print(ev12) ),
/**/End:"Tejun1")$
/*6*/
/*=*/Finished:""$

```

①第1分節は、スクリプトのタイトル

MaximaOnLineではなく、インストール版のmaximaを使う場合の処理方法も指示
 MaximaOnLineでは、「kill(all)」を実行するとグラフが表示されない。
 インストール版では、「kill(all)」を実行しないと、事前に処理したスクリプトの変数値が、
 実行者の意図に反して、処理中のスクリプトに影響を与えるリスクがある。

②第2分節は、課題

「sin(a+30)を加法定理で展開」の提示

③第3分節は、ユーザー定義関数

「sind(t):=sin(t/180*Pi),cosd(t):=cos(t/180*Pi),」の定義。度数法の場合の正弦・余弦関数。

④第4分節は、ユーザー定義定数

「print([Pi:%pi,deg:25/180*Pi]),」の定義と表示。複文である。

⑤第5分節は、処理手順(この分節は、内部をさらにブロック化)

・内部第1ブロックは、この分節の表題と初期値設定

```

(/*=*/ TEJUN1: "1 解析的 sin(a+30)を加法定理で展開すると?",
subl:[a=deg],
print(["subl",subl]) ),

```

出力結果

```

[subl, [a =  $\frac{5\pi}{36}$ ]]

```

・内部第2ブロックは、与式の公式による展開と数値例によるその確認

```

(ev11:["ev11","Ge=sin(a+30)=sin(a)*cos(30)+cos(a)*sin(30)",
/**/float([KKK,subst(subl,[
"Ge: ",sind(a+30), "=",sind(a)*cosd(30)+cosd(a)*sind(30),"(加法定理)" ] ) ] ),
/**/print(ev11) ),

```

出力結果（以降、出力結果では、斜体の部分のみを表示）

```
5 %pi
[a = -----]
36
[ev11, Ge: sin(a+30)=sin(a)*cos(30)+cos(a)*sin(30),
 [KKK, [Ge: , 0.50658059820751, =, 0.50658059820751, (加法定理)]]]
[ev12, =sin(a)*sqrt(3)/2+cos(a)*1/2 =(sqrt(3)*sin(a)+cos(a))/2,
 [KKK, [ =, 0.50658059820751, =, 0.50658059820751, (30°の正弦・余弦)]]]
```

ev11の第1項では、この出力がev11のものであることを明示

第2項では、手計算の実行を記述

第3項では、

floatにより、直後の()内側のもの（引数という）を小数表示し、

この小数表示の一致により、アプリの計算と手計算が、一致しているか確認

引数である外側2番目のリストの第2項は、変数を第1文の初期値設定して、

与式(Ge)の左辺の値(最内側のリストの第2項)と上の第2項の手計算(最内側リストの第3項)の値を計算し、小数表示(手計算が正しければ一致)

・内部第3ブロックは、展開式へ30°の正弦・余弦を代入し、数値例により確認

```
(ev12:["ev12", "=sin(a)*sqrt(3)/2+cos(a)*1/2 =(sqrt(3)*sin(a)+cos(a))/2",
 /**/float([KKK,subst(subl, [
 "=",sind(a)*sqrt(3)/2+cosd(a)*1/2 , "=",(sqrt(3)*sind(a)+cosd(a))/2,"(30°の正弦・余弦)" ] ) ] ),
 print(ev12) ),
```

出力結果

```
[ev12, =sin(a)*sqrt(3)/2+cos(a)*1/2 =(sqrt(3)*sin(a)+cos(a))/2,
 [KKK, [ =, 0.50658059820751, =, 0.50658059820751, (30°の正弦・余弦)]]]
```

ev12の第1項では、この出力がev12のものであることを明示

第2項では、手計算の実行を記述

第3項では、

floatにより、直後の()内側のもの（引数という）を小数表示し、

この小数表示が、2つとも、ev11のものと一致しているかどうかにより、

手計算が正しく進行しているか確認

⑥第6分節は、アプリ終了の表示

2) 分節化スクリプトの修正例（余弦定理の証明）

前節に示した分節化スクリプトを修正して、

余弦定理の証明に利用した例を示す。

最終的に、以下の赤の部分のように書き換えることになる。

なお、具体的な作業手順は、3)に示す。

/* 余弦定理 */

/*(kill(all), print("インストールMaximaでは、行頭の/*を、この行末の「\$」の後にCut&Paste"))\$ */

/*2*/

```

(/*=*/ MONDAI:"三角形CABにおいて、 CA=b, AB=c, BC=aとすると、
 $c^2=a^2+b^2-2ab\cos(C)$ を示せ。(度数法)",
print(MONDAI),
/**/End:"Mondai")$
/*3*/
(KANSUU:"",
sind(t):=sin(t/180*Pi),cosd(t):=cos(t/180*Pi),
/**/End:"Kansuu")$
/*4*/
(TEISUU:"",
print([Pi:%pi,deg:25/180*Pi]),
/**/End:"Teisuu")$
/*5*/
((/*=*/ TEJUN1: "1 解析的  $t_{CAB} (CA=b, AB=c, BC=a) \rightarrow$ 
 $c^2=a^2+b^2-2ab\cos(C)$  ",
print(TEJUN1)),
subl:[C=deg,a=1,b=3/7],
print(["subl",subl]),
(ev11:["ev11","C:鋭角,  $b>a$ の場合, 頂点Bから辺CAに垂線BDを引く。"],
/**/print(ev11) ),
(ev12:["ev12","rhs(Ge) = $a^2 + b^2 - 2ab\cos(C) = a^2 + (b-a\cos(C))^2 - a^2\cos(C)^2$ ",
/**/float([KKK,subst(subl,[
"rhs(Ge) =", $a^2 + b^2 - 2ab\cosd(C)$ ,"=", $a^2 + (b-a\cosd(C))^2 - a^2\cosd(C)^2$ ,"(2項,3項の平方
完成) ] ) ] ),
/**/print(ev12) ),
(ev13:["ev13","= $a^2(1-\cos(C)^2)+(b-a\cos(C))^2$ ",
/**/float([KKK,subst(subl,[
"=", $a^2(1-\cosd(C)^2)+(b-a\cosd(C))^2$ ,"(1項,3項で $a^2$ を括りだし) ] ) ] ),
/**/print(ev13) ),
(ev14:["ev14","= $a^2\sin(C)^2+(b-a\cos(C))^2 = BD^2 + DA^2 = AB^2 = c^2 = lhs(Ge)$ ",
/**/float([KKK,subst(subl,[
"=", $a^2\sin d(C)^2 + (b-a\cosd(C))^2$ ,"=  $BD^2 + DA^2 = AB^2 = c^2 = lhs(Ge)$ ","(正弦・余弦の平
方和と図(略)から) ] ) ] ),
/**/print(ev14) ),
/**/End:"Tejun1")$
/*6*/
/*=*/Finished:""$

```

3) スクリプト編集効率化の工夫

① コメント符号を工夫し、第1文節第1文から順に編集

i) 第2文節以降をコメントアウトし、実質1文の第1文節を編集

第1文節と第2文節の間に、「/*=」と書き込み、最終行の先頭の「/*=」を消す。

その後、「ctrl+f」と打ち、表示された入力枠に「/*=」と打ち、「enter」を1回打って(以降、文修正箇所検索という)、表示された部分にあるタイトルを書換える。その後、「tab」を打ち「click」をクリックして(以降、maxima実行という)、出力確認し、次に進む。

ア) タイトルの書換え

```
/* 余弦定理 */
```

```
/*(kill(all), print("インストールMaximaでは、行頭の/*を、この行末の「$」の後にCut&Paste"))$ */
```

ii) 第3文節以降をコメントアウトし、実質1文の第2文節を編集

第1文節と第2文節の間の「/*=」を消し、第2文節と第3文節の間に、「/*=」と書き込み、その後、文修正箇所検索し、「MONDAI:...」を書換える。

その後、maxima実行して、結果を確認し次に進む。

ア) 「MONDAI:...」の書換え

```
(/*=*/ MONDAI:"三角形CABにおいて、CA=b, AB=c, BC=aとすると、  
c^2=a^2+b^2-2*a*b*cos(C)を示せ。(度数法)",  
print(MONDAI),  
/**/End:"Mondai")$
```

出力結果

```
三角形CABにおいて、CA=b, AB=c, BC=aとすると、c^2=a^2+b^2-2*a*b*cos(C)  
s(C)を示せ。(度数法)
```

iii) 実質最後で実質4文の第5文節を編集

第2文節と第3文節の間の「/*=」を消し、第6文節の先頭に「/*=」と書き込み、修正箇所検索する。その結果、表示される「TEJUN1 解析的 ...」のところの問題を書換え、「print(...)」を追加し、「subl:...」を書換えて、次行の「print(subl),」の後に、「/*=」を書き込み、第5文節の最終行の「/**/End:"Tejun1")\$」の先頭の「/*」を消す(以降、文節内修正箇所検索準備という。)

その後、maxima実行して、結果を確認し、次に進む。

これが、文節中の文を第1文から順に書換・調整する方法である。

ア) 問題の書換え

```
((/*=*/ TEJUN1: "1 解析的 tCAB (CA=b, AB=c, BC=a) ->  
c^2=a^2+b^2-2*a*b*cos (C)",
```

イ) 「print(...)」を追加

```
print(TEJUN1),
```

ウ) 「subl:...」を書換え

```
subl:[C=deg,a=1,b=3/7],
```

maxima実行・結果確認の後、文節内修正箇所検索準備をして、「ev11:...」の書換えに進む。

この段階での、出力結果

```
1 解析的 tCAB (CA=b,AB=c,BC=a) -> c^2=a^2+b^2-2*a*b*cos(C)
          5 %pi          3
[subl, [C = ----, a = 1, b = -]]
          36          7
```

エ) 「ev11:...」の第1,2項を書換え、第3項を削除

```
(ev11:["ev11","C:鋭角, b>=aの場合, 頂点Bから辺CAに垂線BDを引く。"],
/**/print(ev11)),
```

この段階での、出力結果

```
[ev11, C:鋭角, b>=aの場合, 頂点Bから辺CAに垂線BDを引く。]
```

maxima実行・結果確認の後、文節内修正箇所検索準備をして、「ev12:...」の書換えに進む。

オ) 「ev12:...」の書換え

```
(ev12:["ev12","rhs(Ge) =a^2 +b^2 -2*a*b*cos(C)= a^2 +(b-a*cos(C))^2-a^2*cos(C)^2",
/**/float([KKK,subst(subl,[
"rhs(Ge) =",a^2 +b^2 -2*a*b*cosd(C),"=",a^2 +(b-a*cosd(C))^2 -a^2*cosd(C)^2,"(2項,3項の平方
完成)" ] ) ])],
/**/print(ev12)),
```

この段階での、出力結果

```
[ev12, rhs(Ge) =a^2 +b^2 -2*a*b*cos(C)= a^2 +(b-a*cos(C))^2-a^2*cos(C)^2,
[KKK, [rhs(Ge) =, 0.32655546706382, =, 0.32655546706382,
(2項,3項の平方完成)]]]
```

maxima実行・結果確認の後、「ev13:...」の挿入に進む。

カ) 「ev13:...」の挿入

「ev12:」から、次の「print(ev12)」までを、コピーし、「print(ev12)」の次の行に挿入する。
そして、文節内修正箇所検索準備をして、挿入した文を、次のように書換える。

```
(ev13:["ev13","=a^2*(1-cos(C)^2)+(b-a*cos(C))^2",
/**/float([KKK,subst(subl,[
"=", a^2*(1-cosd(C)^2)+(b-a*cosd(C))^2,"(1項,3項でa^2を括りだし)" ] ) ])],
/**/print(ev13)),
```

この段階での、出力結果

```
[ev13, =a^2*(1-cos(C)^2)+(b-a*cos(C))^2,
[KKK, [ =, 0.32655546706382, (1項,3項でa^2を括りだし)]]]
```

maxima実行・結果確認の後、「ev14:...」の挿入に進む。

キ) 「ev14:...」の挿入

「ev13:」から、次の「print(ev13)」までを、コピーし、「print(ev13)」の次の行に挿入する。
そして、文節内修正箇所検索準備をして、挿入した文を、次のように書換える。

```
ev14:["ev14"," =a^2*sin(C)^2+(b-a*cos(C))^2= BD^2+ DA^2= AB^2= c^2= lhs(Ge)",  
/**/float([KKK,subst(subl,[  
"=", a^2*sind(C)^2 +(b-a*cosd(C))^2 ,"= BD^2+ DA^2= AB^2= c^2= lhs(Ge)","(正弦・余弦の平方和と図（略）から" ] ) ] ),  
/**/print(ev14),
```

この段階での、出力結果

```
[ev14, =a^2*sin(C)^2+(b-a*cos(C))^2= BD^2+ DA^2= AB^2= c^2= lhs(Ge),  
[KKK, [ =, 0.32655546706382, = BD^2+ DA^2= AB^2= c^2= lhs(Ge),  
(正弦・余弦の平方和と図（略）から) ] ] ]
```

maxima実行・結果確認の後、「print(ev14)」の後の「/=」を消し、

「*/End:"Tejun1"」の先頭に「/=」を挿入し、maxima実行・結果確認して、編集が終了する。

4) スクリプト編集効率化の意義

- ① 構造が明確で、見通しがよい
- ② 文節ごと、分節内の文ごとにインタープリタ風に実行可能
- ③ ②により、MaximaOnLineのエラーメッセージの貧弱さに対処可能
- ④ 文出力の前半は、手書き解答風、後半は、パソコン計算と手計算の比較・検討
- ⑤ ④により、別途のコマンド学習の必要性減少

5. 代数的・解析的解、作図、図形的解の3要素を備えたスクリプト

1) 分節化スクリプトスタイルの現段階

このスクリプト例は、代数的・解析的解、作図、図形的解の3要素を備え、導き出した解を多面的に考察するためのものである。

いわば、数学的に定義された世界をスクリプトにより仮想的に構成し、グラフ上に再現して、手書き風の解答と仮想的世界での値が一致することを確認しようとするものである。

2) 3要素を備えたスクリプト例

グラフ・図を描画するためのユーザ定義関数を組み込んでいるため、かなり長いスクリプトになっているが、そこそこの汎用性を発揮できるように定義してある。

なお、図中の文字位置の調整は、グラフ準備のdA等による。Aの位置からの相対位置（ベクトル）をdAで与えるようにしてある。現実的には、試行的に出力したグラフ・図を見ながら、相対位置 d A等を調整すればよい。

```
/*WORK:"502 余弦定理●●●●"*/
```

```

/*(kill(all),CHECK:"(インストールMaximaでは、行頭の/*を行末へ移動すること)")$
*/
/*2*/
(MONDAI:"● 三角形CABにおいて、 $a\cos B=a\cos C$ ,  $CA=b$ ,  $AB=c$ ,  $BC=a$ とすると、
 $c^2=a^2+b^2-2ab\cos(C)$ を示せ。" /*(◆変更部分)*/,
KOUSATU:"tCAB,aACB=aC,CA=b,BC=a,sbl:[aC=60,b=3,a=2]として、
 $c^2=a^2+b^2-2ab\cos(aC)$ " /*(◆変更部分)*/,
End:"Mondai")$
/*3*/
(KANSUU:"",
gr(T,c,l,pB,pCR,lab,xm,xM,ym,yM):=block(
/**/wxdraw2d(
/*...*/title =T, yrange=[ym,yM] /* "タイトルと値域" */,
/*...*/implicit(c,x,xm,xM,y,ym,yM),
/*...*/color=black,
/*...*/implicit(l,x,xm,xM,y,ym,yM),
/*...*/implicit(y=0,x,xm,xM,y,ym,yM),implicit(x=0,x,xm,xM,y,ym,yM) /* "x,y軸" */,
/*...*/color=red,point_type =7,line_width=2,
/*...*/points_joined =true,point_size =1,points(pCR[2]) /* "点pCR[2]を赤で" */,
/*...*/color=cyan,point_type =7,line_width=2,
/*...*/points_joined =true,point_size =1,points(pCR[1]) /* "点pCR[1]をシアンで" */,
/*...*/color=blue,point_type =6,
/*...*/points_joined =true,point_size =1,points(pB) /* "点pBを青で" */,
/*...*/color=red,lab[2][1],lab[2][2],lab[2][3],lab[2][4],lab[2][5],lab[2][6],lab[2][7],
/*...*/lab[2][8],lab[2][9],lab[2][10],lab[2][11],lab[2][12],lab[2][13],lab[2][14],lab[2][15],
/*...*/lab[2][16],lab[2][17],lab[2][18],lab[2][19]) /* "文字書込" */,
lab(nam,p):=label([nam,p[1],p[2]]) /* 図中文字 */,
ng2pl(p1l,p2l,m,n):=(p1l*n+p2l*m)/(m+n) /*2点と比から内外分点*/,
p2ngl(pl,p1l,p2l):=block([P,p1,p2,ans],
/**/p1:pl-p1l,p2:pl-p2l,P:rat(sqrt(p1.p1)/sqrt(p2.p2)),p1:num(P),p2:denom(P),
/**/print(ng2pl(p1l,p2l,p1,p2)),mes:"内分",
/**/if ng2pl(p1l,p2l,p1,p2)#pl then ((if p1>p2 then p2:-p2 else p1:-p1),mes:"外分"),
/**/ans:[p1,".",p2,mes] /*点を2点の内外分比で表示*/),
xtpl(p1l):=block([ans],ansl:[p1l[1],-p1l[2]]) /*x軸対称*/,
ytpl(p1l):=block([ans],ansl:[-p1l[1],p1l[2]]) /*y軸対称*/,
otpl(p1l):=block([ans],ansl:[-p1l[1],-p1l[2]]) /*原点对称*/,
a12(p1l,p2l):=block([ans],if p2l[1]#p1l[1] then ans:(p2l[2]-p1l[2])/(p2l[1]-p1l[1])
/**/else ans:9999 /*傾き*/),
gl(p1l,p2l):=block([ans],ansl:[p2l[1]-p1l[1],p2l[2]-p1l[2]]) /*傾き方向*/,
nl(p1l,p2l):=block([ans],ansl:[p2l[2]-p1l[2],-(p2l[1]-p1l[1])],
/**/ansl:ansl/gcd(ansl[1],ansl[2]) /*垂直方向*/),
dis(p1l,p2l):=block([ans],ans:float(sqrt((p2l-p1l).(p2l-p1l)))) /*距離*/,
pppl(p1l,p2l,p3l):=block([ans],ansl:((p1l-p2l).(p3l-p2l)/dis(p3l,p2l))*(p3l-p2l)/dis(p3l,p2l)),

```

```

sind(t):=sin(t/180*%pi),
cosd(t):=cos(t/180*%pi),
tand(t):=tan(t/180*%pi),
asind(sind):=asin(sind)/%pi*180,
acosd(cosd):=acos(cosd)/%pi*180,
atand(tand):=atan(tand)/%pi*180,
a(p1l,p2l,p3l):=
/**/float(acosd((dis(p1l,p2l)^2+dis(p2l,p3l)^2-dis(p1l,p3l)^2)/(2*dis(p1l,p2l)*dis(p2l,p3l)))),
/**/End:"Kansuu")$
/*4*/
(TEISUU:"",
Pi:%pi,
deg:25/180*Pi/**/,
End:"Teisuu")$
/*5*/
(TEJUN1: "1 解析的",
ex10:["ex10, tCAB,aBCA=aC,CA=b,BC=aとして,BからCAに垂線BDを下ろして、
c^2=a^2+b^2-2*a*b*cos(aC)",sbl:[aC=60,b=3,a=2],
/*=*/float(subst(sbl,[KKK,["tCAB,aBCA=aC=",aC,"CA=b=",b,"BC=a=",a,"c^2=",a^2+b^2-2*a*b*
cosd(aC) ] ] ) ) ) /* (◆修正) *//,
print(ex10),
ex11:["ex11, RHS(Ge)=a^2+b^2-2*a*b*cos(aC)=a^2+(b-a*cos(aC) )^2-a^2*cos(aC)^2] << 余弦
定理 >>",
/**/float(subst(sbl,[KKK,[" =",c2:a^2+b^2-2*a*b*cosd(aC)," =",a^2*(1-cosd(aC)^2)
+(b-a*cosd(aC) )^2 ] ] ) ) /* (◆修正) *//,
print(ex11),
ex12:["ex12, =a^2*(1-cos(aC)^2) +(b-a*cos(aC) )^2 =a^2*sin(aC)^2 +(b-a*cos(aC) )^2 << a^2の
括りだし,正弦・余弦の平方和 >>",
/**/float(subst(sbl,[KKK,[" =",a^2*(1-cosd(aC)^2) +(b-a*cosd(aC) )^2, " =",a^2*sind(aC)^2
+(b-a*cosd(aC) )^2 ] ] ) ) /* (◆修正) *//,
print(ex12),
ex13:["ex13, =BD^2 +DA^2 =BA^2 =c^2 =RHS(Ge) << 直角三角形ADB >>",
/**/float(subst(sbl,[KKK,[" =BD^2 +DA^2 =BA^2 =",c2," =RHS(Ge)" ] ] ) ) /* (◆修正) *//,
print(ex13),
/**/End:"Tejun1")$
/*6*/
(GJUNBI:"グラフ準備",print(["tCAB,aBCA=aC,CA=b,BC=a,sbl:[aC=",aC:60,"b=",b:3,"a=",a:2,"]
として、c^2=a^2+b^2-2*a*b*cos(aC)",r=",r:3]) /* (◆修正) *//,
A:[b,0],B:a*[cosd(aC),sind(aC)],C:[0,0],D:[a*cosd(aC),0] /* (◆修正) *//,
dA:[-0.1,-0.1]*r, dB:[0.1,0.1]*r, dC:[-0.1,0.1]*r, dD:[0.1,0.1]*r ,
labA:lab("A",A+dA), labB:lab("B",B+dB), labC:lab("C",C+dC),labD:lab("D",D+dD) ,
print(float([labA,labB,labC,labD] ) ) ,
/**/End:"GJunbi")$

```



```

/*7*/
(GKAKUNIN:"グラフを書いて、視覚的に確認",
yL:r*1.2,xL:yL/4.8*6.8,
c0:y=0,
T:"For  $c^2 = a^2 + b^2 - 2ab \cos(aC)$ " /* (◆修正) */,
O:[xL,yL],dO:[0,0],labO:lab(" ",O+dO),
lab:[19,[labO,labA,labB,labC,labD,labO,labO,labO,labO,labO,labO,labO,labO,labO,labO,labO],
/**/labO,labO,labO,labO,labO]] ,
pCR:[[B,C,A],[A,B]] /* (◆修正) */,
pB:[B,C,D,B] /* (◆修正) */,
print(float([gk10,[T,c0,l:x=0,pB,pCR,lab,-xL,xL,-yL,yL] ] ) /**/,
gr(T,c0,l,pB,pCR,lab,-xL,xL,-yL,yL) /* "グラフ出力すると" */ /**/,
/**/End:"GKakunin")$
/*8*/
(TEJUN2: "2 図形的",
print(["tCAB,aBCA=aC=",aC:60,"CA=b=",CA:3,b:3,"BC=a=",BC:2,a:2," $c^2 = a^2 + b^2 - 2ab \cos(aC)$  ? "] ) /* (◆修正) */,
ex21:["ex21", "LHS(Ge) =  $c^2 = BD^2 + DA^2 = (BC \sin(aC))^2 + (CA - BC \cos(aC))^2 = a^2 \sin(aC)^2 + (b^2 - 2ab \cos(aC) + a^2 \cos(aC)^2$  << tADB:直角三角形 >>",
/**/float([KKK,subst(sbl,["LHS(Ge) =  $c^2 =$ ",dis(B,D)^2 + dis(D,A)^2,"=", (BC*sind(aC))^2 + (CA-BC*cosd(aC))^2,"=",a^2*sind(aC)^2 + (b^2 - 2ab*cosd(aC) + a^2*cosd(aC)^2) ] ) ] ) /* (◆修正) */,
/**/print(ex21),
ex22:["ex22", " =  $a^2 (\sin(aC)^2 + \cos(aC)^2) + b^2 - 2ab \cos(aC) = a^2 + b^2 - 2ab \cos(aC) = \text{RHS(Ge)}$  ", "正弦・余弦の平方和",
/**/float([KKK,subst(sbl,[" =",a^2*(sind(aC)^2 + cosd(aC)^2)+b^2 - 2ab*cosd(aC),"=",a^2 + b^2 - 2ab*cosd(aC)," = RHS(Ge)" ] ) ] ) /* (◆修正) */,
/**/print(ex22),
ans:["LHS(Ge)=RHS(Ge)", "結論",float([subst(sbl,[c^2,"=",dis(A,B)^2,"=",a^2 + b^2 - 2ab*cosd(aC) ] ) ] ) /* (◆修正) */,
/**/print(ans),
/**/End:"Tejun2")$
/*9*/
(MATOME:"",
print(MONDAI),
print(ans),
/**/End:"EndMatome")$
/*10*/
/*=/**/◆◆◆◎上のMaximaコマンドを実行せよ。
◎上のコマンドを、次により修正して実行し、結果を比較して考察せよ。
/**/
(MONDAI:"●例題6・練習2 2 502 △ABCで、 $a=3, b=2\sqrt{2}, C=45^\circ$ のとき、 $c$ は" /* (◆変更部分) */,

```

KOUSATU:"~三角形は、△CBA,CB=a,aBCA=aC,AC=b,sbl:[a=3,b=2*sqrt(2),aC=45]として、AB
 は?"/*(◆変更部分)*/
 /**/
 */END:"Finish"\$

出力結果（斜体の部分のみ）

TEJUN1: "1 解析的",

```
[ex10, tCAB,aBCA=aC,CA=b,BC=aとして,BからCAに垂線BDを下ろして、c\
^2=a^2+b^2-2*a*b*cos(aC), [aC = 60, b = 3, a = 2],
[KKK, [tCAB,aBCA=aC=, 60.0, CA=b=, 3.0, BC=a=, 2.0, c^2=, 7.0]]]
[ex11, RHS(Ge)=a^2+b^2-2*a*b*cos(aC)=a^2+(b-a*cos(aC))^2-a^2*cos(aC)^2] << ◆\
◆弦定理 >>, [KKK, [ =, 7.0, =, 7.0]]]
[ex12, =a^2*(1-cos(aC)^2) + (b-a*cos(aC))^2 =a^2*sin(aC)^2 + (b-a*cos(aC))^2 <\
< a^2の折りだし,正弦・余弦の平方和 >>, [KKK, [ =, 7.0, =, 7.0]]]

[ex13, =BD^2 +DA^2 =BA^2 =c^2 =RHS(Ge) << 直角三角形ADB >>,
[KKK, [ =BD^2 +DA^2 =BA^2 =, 7.0, =RHS(Ge)]]]
```

コメント

ex10~ex13の各文の前半は手書き解答、後半（KKK以降）の数値はアプリ内の値で、手書き解
 答の値と突き合わせ確認

全角文字列の途中で改行が生じると、判読不明な記号が表示される。

(GJUNBI: "グラフ準備",

```
[tCAB,aBCA=aC,CA=b,BC=a,sbl:[aC=, 60, b=, 3, a=, 2,
]として、c^2=a^2+b^2-2*a*b*cos(aC), r=, 3]
[label([A, 2.7, - 0.3]), label([B, 1.3, 2.032050807568878]),
label([C, - 0.3, 0.3]), label([D, 1.3, 0.3])]
```

コメント

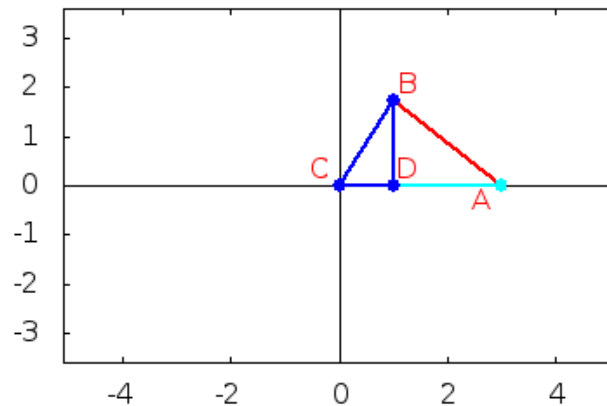
最初のgj10は問題設定の確認で、アプリ内の変数値には関係しない

次のgj11はグラフ内の文字とその表示位置の数値確認で、表示位置が変数のままだと作図処理
 中でエラーが発生する。

(GKAKUNIN: "グラフを書いて、視覚的に確認",

```
[gk10, [For c^2 = a^2 + b^2 - 2*a*b*cos(aC), y = 0.0, x = 0.0,
[[1.0, 1.732050807568877], [0.0, 0.0], [1.0, 0.0], [1.0, 1.732050807568877]],
[[[1.0, 1.732050807568877], [0.0, 0.0], [3.0, 0.0]],
[[3.0, 0.0], [1.0, 1.732050807568877]]],
[19.0, [label([ , 5.1, 3.6]), label([A, 2.7, - 0.3]),
label([B, 1.3, 2.032050807568878]), label([C, - 0.3, 0.3]),
label([D, 1.3, 0.3]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6]),
label([ , 5.1, 3.6]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6]),
label([ , 5.1, 3.6]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6]),
label([ , 5.1, 3.6]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6]),
label([ , 5.1, 3.6]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6]),
label([ , 5.1, 3.6]), label([ , 5.1, 3.6]), label([ , 5.1, 3.6])], - 5.1,
5.1, - 3.6, 3.6]]
```

For $c^2 = a^2 + b^2 - 2*a*b*cos(aC)$



コメント

gk10はグラフ描画関数grの引数の数値確認用である。関数式の係数や座標が変数のままだと作図処理中でエラーが発生する。

グラフの上部は、グラフのタイトルで、1バイト系の文字と数字しか表示できない。

(TEJUN2: "2 図形的",

```
[ex20, tCAB,aBCA=aC=, 60, CA=b=, 3, 3, BC=a=, 2, 2,
c^2 = a^2 + b^2 - 2*a*b*cos(aC) ?]
[ex21, LHS(Ge) = c^2 = BD^2 + DA^2 = (BC*sin(aC))^2 + (CA-BC*cos(aC))^2 = a^2*sin(\
aC)^2 + (b^2 - 2*a*b*cos(aC) + a^2*cos(aC)^2 << tADB:直角三角形 >>,
[KKK, [LHS(Ge) = c^2 =, 7.0, =, 7.0, =, 7.0]]]
[ex22, = a^2*(sin(aC)^2 + cos(aC)^2) + b^2 - 2*a*b*cos(aC) = a^2 + b^2 - 2*a*b*cos(aC\
) = RHS(Ge) , 正弦・余弦の平方和,
[KKK, [ =, 7.0, =, 7.0, = RHS(Ge) ]]]
[LHS(Ge)=RHS(Ge), 結論, [[c^2 =, 7.000000000000001, =, 7.0]]]
```

コメント

ex20の末尾の「？」で、半角の「？」を使うとエラーが生じる。

「結論」という項のある行の上の行にポツンと「2」が表示されているが、これは、元来「 c^2 」を表現する右肩の「2」である。全角の「結論」が表示されたため、ズレが生じている。

「7.000000000000001」はスクリプトが創出した図形平面における2点A,Bの距離の平方であり、「7.0」は余弦定理で計算された c^2 の値である。

```
(MATOME: "",
```

```
    • 三角形CABにおいて、 $a \angle C = a \angle C$ ,  $CA = b$ ,  $AB = c$ ,  $BC = a$ とすると、 $c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos(C)$ を示せ。  
    [LHS (Ge)=RHS (Ge), 結論, [[c , =, 7.000000000000001, =, 7.0]]]
```

コメント

まとめにおいて、課題と結論が簡潔に繰り返される。

printコマンドでの出力で、自動的に改行される部分がascii文字なら、「\」で改行を明示して、改行を行われる。

6. まとめ

1) 分節化3要素スクリプトスタイルの到達点

① スクリプトにより仮想世界を構成

前節5.の例にしたがって説明を加えると、このスクリプトの枠組みを活用して、課題として提起された世界をスクリプトにより仮想的に構成し、グラフ上に再現

② 仮想世界の可視化から解決への筋道を展望

グラフ上に再現した仮想世界の構造において、解決の方向性を展望し、その筋道を複数構想し、比較検討

③ 筋道を確認しながら着実に解決へ

条件を確認しながら、解決への道を一步一步前進

解決の筋道を手書き風の解答と仮想的な世界での値が一致することを確認しながら前進

2) 今後の課題

① 集合・論理的分野、統計・確率的分野のスクリプトスタイル

ほぼ確定できた代数的、解析的、図形的分野に対し、集合・論理的分野、統計・確率的分野のスクリプトスタイルの具体化が残る課題である。

② 数学的活動の取組として

授業前学習用の教材として準備してきたが、2018年に告示された高等学校学習指導要領にいう数学的活動の一環として、生徒自身がスクリプトを作成することも可能である。この方面の取組も今後の課題としたい。