

K_{ET}Cindy で教材作成

久保 康幸 *・高遠 節夫 **

Creating Some Teaching Materials with K_{ET}Cindy

Yasuyuki Kubo *, Setsuo Takato **

Abstract

In this paper, We introduce how to make some teaching materials with K_{ET}Cindy.

1. はじめに

K_{ET}Cindy は、動的幾何ソフト Cinderella を GUI として利用し、K_{ET}Pic を動かす、正確な図入り教材を作成するシステムであり、現在は、色々な外部プログラムを利用してもっと便利で正確に教材作成ができるように開発が進んでいる。

ここでは、Cinderella との連携部分を利用して教材を作る場面において気がついたことを紹介する。なお、紹介する命令文は断らない限り、Cindyscript の作図に関する命令文であり、行を判別するため、命令の各行の先頭に行番号を付けた。

2. 正三角形を含む図

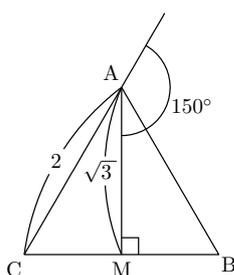


図 1 正三角形を含む図

```
01:Plotdata("1", "sqrt(3)*x", "x=[0,4]");
02:Putpoint("A", [2,2*sqrt(3)]);
03:Putpoint("B", [4,0]);
04:Putpoint("C", [0,0]);
05:Letter([A,"n2w", "A",B,"se", "B",C,
"s2w", "C"]);
06:Listplot([A,B,C]);
```

```
07:Letter(D, "s2", "M");
08:Putpoint("E", [3,3*sqrt(3)]);
09:Listplot([A,D]);
10:Paramark([A,D,B], .7);
11:Anglemark([D,A,E], [2, "Expr=1.5, 150
~{\circ}"]);
12:Bowdata([A,C], [1, .6, "Expr=2"]);
13:Bowdata([A,D], [1, .6, "Expr=\sqrt{3}>
\>>"]);
```

完成図に、4点 A,B,C,M のみ現れるが、作図のために直線 CA 上に点 E を置いてある。また、作図では、点 D として扱ったものを点 M と表示してある。後で大きさや角度を変更しない図のため点 A,B,C を座標で指定した。点 D は、Scinderella 上で B,C の中点として定義したのでスクリプトに点 D の定義が現れない。

3. 平行四辺形とベクトル

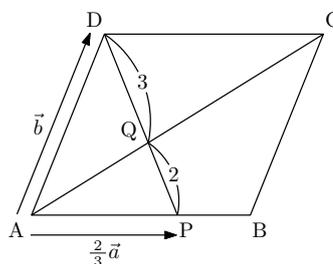


図 2 平行四辺形とベクトル

```
01:Putpoint("A", [0,0]);
02:Putpoint("B", [6,0]);
03:Putpoint("D", [A.x+C.x-B.x,C.y]);
```

```

04:Putpoint("F",[E.x+D.x-A.x,E.y+D.y
-A.y]);
05:Putpoint("H",[G.x+P.x-A.x,G.y]);
06:Putpoint("P",[2*B.x/3,0]);
07:Putpoint("Q",[ (2*D.x+3*P.x)/5,2
*D.y/5]);
08:Letter([A,"s3w3","A",B,"s3e","B"]);
09:Letter([C,"n3e","C",D,"n3w","D"]);
10:Letter([Q,"nw5","Q",P,"s3e","P"]);
11:Listplot([A,B,C,D,A,C]);
12:Listplot([D,P]);
13:Bowdata([P,Q],[1.5,.6,"Expr=2"]);
14:Bowdata([Q,D],[1.5,.6,"Expr=3"]);
15:Arrowdata([E,F]);
16:Arrowdata([G,H]);
17:Expr([[ (E.x+F.x)/2,(E.y+F.y)/2],"w5",
"\vec{b}"]);
18:Expr([[ (G.x+H.x)/2,G.y],"s2","\frac{
\>2\>}{3}\vec{a}"]);

```

2点A,Cの座標から点B,Dを定義し平行四辺形を作成した。自由点E,Gを始点として辺AD,ABに平行なベクトルを作成し、完成図をみて位置を調整した。点Qは線分DPの内分点として定義した。

4. 円周角の図

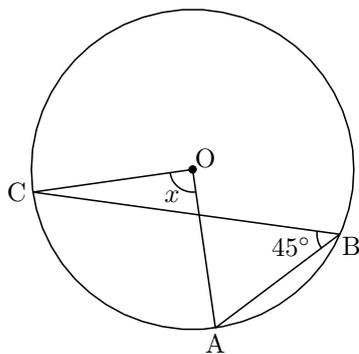


図3 円周角の図

```

01:Putpoint("O",[0,0]);
02:Circledata([O,A]);
03:PutonCurve("B","crOA",[0,"A.x"]);
04:Putpoint("C",[A.y,-A.x]);
05:Listplot([A,B,C,O,A]);
06:Anglemark([C,O,A],[ "Expr=x"]);
07:Anglemark([C,B,A],[ "Expr=2.2,45^\circ"]);

```

```

08:Pointdata("1",0,["size=5"]);
09:Letter([A,"s2","A",B,"se","B",C,"w2",
"C",0,"ne","0"]);

```

中心O(動かさないで命令により定義)と円周上の1点A(Cinderella上で自由点として定義)を取り、02行目のCircledata([O,A]);により円周を定義して、円周上に2点B,Cを取っている。点Cは点Aを時計回りに90°回転させて定義した。点(a,b)を原点中心に-90°回転させた点の座標が(-b,a)であるという性質を利用している。点描画された図を見ながら点を動かして見やすい位置に3点を調整した。Cinderella上で点を動かすとき、円周角45°が維持され、また、Anglemarkコマンドで角度マークを描くので、見やすい図に調整するのに、点Aと点Bをマウスで動かすことに専念できる。

5. 三角形の頂点を座標で示す図

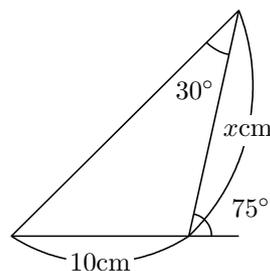


図4 三角形の頂点を座標で示す図

```

01:PutPoint("A",[1,1],[A.x,A.x]);
02:PutPoint("C",[A.x*(sqrt(3)-1),0]);
03:PutPoint("D",[A.x,0]);
04:Listplot([D,B,A,C]);
05:Bowdata([C,A],[1.5,.7,"Expr=x\text{cm}"]);
06:Bowdata([B,C],[1.5,1.5,"Expr=10\text{cm}"]);
07:Anglemark([B,A,C],[2,"Expr=2,30^\circ"]);
08:Anglemark([D,C,A],[ "Expr=2,\hspace{7mm}75^\circ"]);

```

Cinderella上の作図より座標の計算が早かったので上のようにした。Cinderellaの作図機能をメインとして同じ図を描くこともできる。手順としては、まず、次のようにCinderella上で図を描く。

(1) 2点A, Bを与え、直線ABを引く。

(2) 中心を A とし点 B を通る円と、中心を B とし点 A を通る円を描き、それらの交点を C とする。

(3) 直線 AB の垂線で点 C を通るものを描き、垂線の足を点 E とする。

(4) $\angle ACE$ の二等分線を加え、直線 AB との交点を点 F とする。

(5) 中心を E とし点 C を通る円を描き、直線 AB との交点を点 H とする。

次に点 E,H,C,F を順に結び、長さを示す弧などを加える。スクリプトは次のようにすればよい。

```
01:Listplot([E,H,C,F]);
02:Bowdata([F,C],[1.5,.7,"Expr=x\text{cm}"]);
03:Bowdata([H,F],[1.5,1.5,"Expr=10\text{cm}"]);
04:Anglemark([H,C,F],[2,"Expr=2,30^\circ"]);
05:Anglemark([E,F,C],[1,"Expr=2,\hspace{7mm}75^\circ"]);
```

6. 2つの図を重ねる

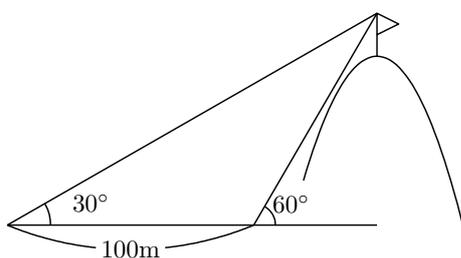


図5 黒い図

```
01:PutPoint("A",[1,1],[A.x,A.x/sqrt(3)]);
02:PutPoint("C",[A.x*(2/3),0]);
03:PutPoint("D",[A.x,0]);PutPoint("E",
[A.x,A.y-.5]);
04:PutPoint("F",[A.x+.5,A.y-.25]);PutPoint
("G",[A.x,A.y-1]);
05:Listplot([E,F,A,G]);
06:Listplot([A,B]);
07:Listplot([A,C]);
08:Listplot([B,D]);
09://Listplot([G,D]);
10://Paramark([C,D,G]);
11:Plotdata("1","-(x-A.x)^2+G.y","x=
[A.x-1.7,A.x+2]");
12:Bowdata([B,C],[1,1.6,"Expr=100\text{m}"]);
```

```
13://Bowdata([C,D],[1,1,"Expr=x\text{m}
"]);
14://Bowdata([B,D],[1.5,1,"Expr=y\text{
}{m}"]);
15://Bowdata([D,A],[1,1,"Expr=h\text{m}
"]);
16:Anglemark([C,B,A],[2,"Expr=2,30^\circ
"]);
17:Anglemark([D,C,A],[1,"Expr=2,60^\circ
"]);
```

先頭が//となっている行は、コメントアウトした命令文である。これに、次の図を重ねる。

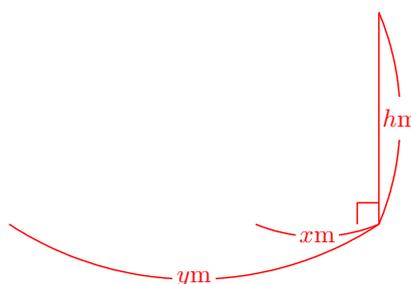


図6 赤い図

```
01:PutPoint("A",[1,1],[A.x,A.x/
sqrt(3)]);
02:PutPoint("C",[A.x*(2/3),0]);
03:PutPoint("D",[A.x,0]);PutPoint("E",
[A.x,A.y-.5]);
04:PutPoint("F",[A.x+.5,A.y-.25]);
PutPoint("G",[A.x,A.y-1]);
05://Listplot([E,F,A,G]);
06://Listplot([A,B]);
07://Listplot([A,C]);
08://Listplot([B,D]);
09:Listplot([A,D]);
10:Paramark([C,D,G]);
11://Plotdata("1","-(x-A.x)^2+G.y","x=
[A.x-1.7,A.x+2]");
12://Bowdata([B,C],[1,1.6,"Expr=100
\text{m}"]);
13:Bowdata([C,D],[1,1,"Expr=x
\text{m}"]);
14:Bowdata([B,D],[1.5,1,"Expr=y
\text{m}"]);
15:Bowdata([D,A],[1,1,"Expr=h
\text{m}"]);
```

```
16://Anglemark([C,B,A],[2,"Expr=2,30
^\\circ"]);
17://Anglemark([D,C,A],[2,"Expr=2,60
^\\circ"]);
```

2つの図の命令文を比較すると、初めの4行は同じで、05行目から、コメントアウトを反対にしている。これにより、同じ4つの点から異なる図を描いている。図5と図6を重ねて、次の図7のようにできる。スクリプトにより4点を全く同じ配置にし、ketlayerにより図の位置を座標で指定しているので、正確に、ぴったり重なった図になっている。なお、山に見立てたカーブは、放物線であるが、もっと複雑な曲線にしたければ、 $K\epsilon T C i n d y$ に用意されたベジェ曲線などで描ける。

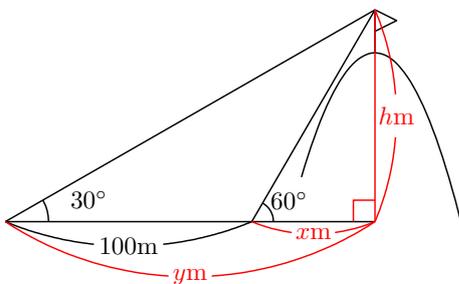


図7 黒い図と赤い図を重ねる

7. 大学入試の解答例から

新聞で紹介された2016年度の愛媛大学2次試験(前期)の解答例が手書きだったのだが、幾つかの図・表を $K\epsilon T C i n d y$ で作成してみた。

7.1 増減表

$K\epsilon T C i n d y$ の作表コマンドには、TabledataおよびTabledatalightがあり、Tabledataでの作表はCinderellaの画面上でインタラクティブに微調整できるが、今回は、スクリプトだけで縦横幅を変更できるTabledatalightを使って増減表を作ってみた。

表1 増減表

x	...	1	...
$f'(x)$	+	0	-
$f(x)$	↗	極大 $\frac{1}{e}$	↘

```
01:xLst=[10,.5,8,10,8];
```

```
02:yLst=[5,5,7];
03:rmvL=["r0c0c5","r3c0c5","c0r0r3","c5r0r3"];
04:Tabledatalight("",xLst,yLst,rmvL);
05:Putcolexpr(1,"c",["x","f'\prime(x)","f(x)"]);
06:Putcolexpr(3,"c",["\cdots","+","\nearrow"]);
07:Putcol(4,"c",["$1$","$0$",""]);
08:Putcol(4,"s",["","","\frac{1}{e}"]);
09:Putcolexpr(5,"c",["\cdots","-","\searrow"]);
10:Putcell(4,3,"t",{"\scriptsize 極大"});
```

01~04行で作表し、05行目以降でセルに文字や記号を入れている。

なお、2重線に見える縦の線は、横幅の小さい列を挿入して実現した。04行の作表コマンドに、縦横幅の数値および(罫線を描かない)除外線のリストを直に入力すれば01~03行は不要だが、指定内容を分かりやすくするため4行に分けて作表した。

また、05~09行のような、表の列を指定したセルに文字を入れるPutcolでは罫線との位置を詰めることができなかったため、また、「極大」という文字を一カ所のセルに追加するだけなので、10行目でPutcellコマンドで利用した。

7.2 グラフで囲まれた領域に斜線を引く(1)

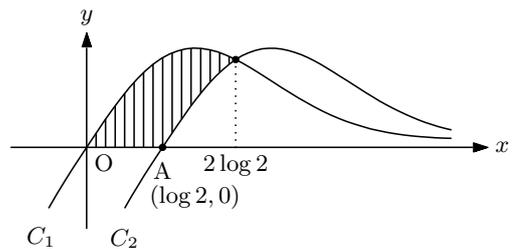


図8 領域に斜線(1)

```
01:Addax(1);
02:Setax(["a","","","","","","se"]);
03:Plotdata("1","(1.5*x*exp(-x*abs(x))+.02*x)","x=[-.25,2.4]");
04:Plotdata("2","(1.5*(x-.5)*exp(-(x-.5)*abs(x-.5))+.02*(x-.5))","x=[.25,2.4]");
05:Plotdata("3","0","x=[-1,A.x]","[notex]");
06:Putpoint("0",[0,0]);
```

```

07:Putintersect("A","gr1","gr2");
08:Putpoint("B",[A.x,0]);
09:Listplot([0,B],["notex"]);
10:Putintersect("C","sg0B","gr2");
11:Listplot([0,C],["notex"]);
12:Listplot([C,B],["notex"]);
13:Listplot([A,B],["do"]);
14:Putpoint("D",[C.x,1],[C.x,D.y]);
15:Pointdata("4",[A,C],["size=4"]);
16:Letter(C,"s1","A");
17:Letter(C,"s4e","$\hspace{-2ex}(\log
  2,0)$");
18:Letter(B,"s","$2\log 2$");
19:Letter(E,"s","$C_1$");
20:Letter(F,"s","$C_2$");
21:if(1==0,
22:Hatchdata("5",["iii"],[["gr1","s"],
  ["gr2","n"],["sgCB","n"]],[90]);
23:Hatchdata("6",["ii"],[["gr1","s"],["
  sg0C","n"]],[90]);
24:);

```

22, 23 行目の Hatchdata コマンドにより動作が重くなるので、21 行目と 24 行目の「if(1==0,」と「);」ではさんで、領域に斜線を引かずに作図する。他の作図が完成後に 21 行目を「if(1==1」に直し、領域に斜線を描画した。

また、領域を 2 つに分けて斜線を引いた。曲線 C_1, C_2 の交点から x 軸へ下ろした垂線の足を点 B とし、曲線 C_1 の下、曲線 C_2 の上、線分 AB の上で指定した範囲、および曲線 C_1 の下、線分 OA の上で指定した範囲を合わせれば、図に示したような斜線の領域になる。作図の補助である線分 OA, AB は描画が不要なので、11, 12 行目の命令で、「notex」オプションを付けている。

7.3 グラフで囲まれた領域に斜線を引く (2)

図 8 を少し直したのが図 9 である。先に、作図内容の説明を紹介しておく。

スクリプトの 01 行目は座標軸を表示しない設定である。前の図のスクリプトで 01,02 行目は座標軸の設定であり、その代わりに、02~06 行で座標軸を作図している。

その意図は、次の通りである。曲線 C_2 のグラフの左端を y 軸より左にすることにより、25 行目のように、ひとまとまりの領域に対して斜線を引くことができる。そこで、26 行目はコメントアウトした。

このとき、曲線 C_2 と y 軸を重ねたくなかったので、グラフを描く領域より狭い範囲に座標軸を描く

ため、この説明の初めに書いたように、いったん 01 行目で座標軸を非表示にしてから作図している。

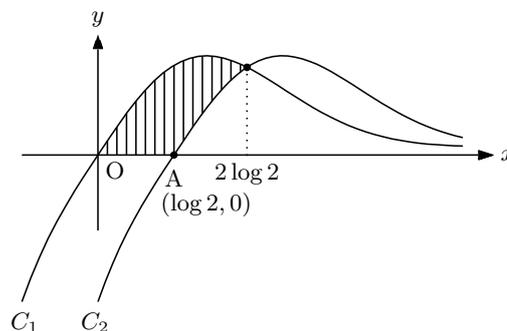


図 9 領域に斜線 (2)

```

01:Addax(0);
02:Arrowdata("ax",[[0,-.5],[0,.8]]);
03:Arrowdata("ay",[[-.5,0],[2.6,0]]);
04:Letter([0,0],"se","0");
05:Letter([0,.8],"n","$y$");
06:Letter([2.6,0],"e","$x$");
07:Plotdata("1","(1.5*x*exp(-x*abs(x))+
  .02*x)","x=[-.5,2.4]");
08:Plotdata("2","(1.5*(x-.5)*exp(-(x-.5)
  *abs(x-.5))+.02*(x-.5))","x=[0,2.4]");
09:Plotdata("3","0","x=[-1,A.x],["notex
  "]);
10:Putpoint("0",[0,0]);
11:Putintersect("A","gr1","gr2");
12:Putpoint("B",[A.x,0]);
13:Listplot([0,B],["notex"]);
14:Putintersect("C","sg0B","gr2");
15:Listplot([0,B],["notex"]);
16:Listplot([A,B],["do"]);
17:Putpoint("D",[C.x,1],[C.x,D.y]);
18:Pointdata("4",[A,C],["size=4"]);
19:Letter(C,"s1","A");
20:Letter(C,"s4e","$\hspace{-2ex}(\log
  2,0)$");
21:Letter(B,"s","$2\log 2$");
22:Letter(E,"s","$C_1$");
23:Letter(F,"s","$C_2$");
24:if(1==0,
25:Hatchdata("5",["iii"],[["gr1","s"],
  ["gr2","n"],["sg0B","n"]],[90]);
26://Hatchdata("6",["ii"],[["gr1","s"],
  ["sg0C","n"]],[90]);
27:);

```

7.4 正多角形の一部

次の図は、4点 O, A_1, A_2, A_3 を結んだのではなく、circledata コマンドを利用した。円の描画は、実は正 n 角形の描画であり、 n を大きくすることにより、円のように見せている。オプションで分割数を 12 にすれば正 12 角形が描ける。また、Rng オプションにより円の一部として扇形を作図できる。

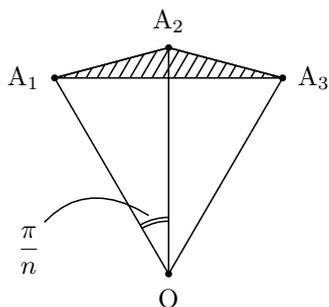


図 10 正多角形の一部

```
01:Circledata("1", [A,B], ["Num=2", "Rng=
[pi/2-pi/6,pi/2+pi/6]"]);
02:Putpoint("C", A.xy+|A,B|*[sin(pi/6),
cos(pi/6)]);
03:Putpoint("D", A.xy+|A,B|*[-sin(pi/6),
cos(pi/6)]);
04:Anglemark("1", [B,A,D], [1.5]);
05:Anglemark("2", [B,A,D], [1.4]);
06:Listplot([A,B]);
07:Listplot([C,A,D]);
08:Pointdata("1", [A,B,C,D], ["size=4"]);
09:Listplot([C,B,D,C]);
10:Letter(A,"s3", "O");
11:Letter(D,"w3", "A$_1$");
12:Letter(B,"n3", "A$_2$");
13:Letter(C,"e3", "A$_3$");
14:Letter(E,"sw", "$\frac{2\pi}{n}$");
15:Bezier("1", [E,F], [G]);
16:if(1==0,
17:Hatchdata("1", ["i"], [{"sgCBDC"}], [6
0]);
18:);
```

まず Cinderella 上で中心とする点 A(表示は O) と円周上の点 B(表示は A_2) を取る。01 行目で、点 A を中心とし、半径 AB の扇形を角度の範囲を $\frac{\pi}{2} - \frac{\pi}{6}$ から $\frac{\pi}{2} + \frac{\pi}{6}$ までとして描く。この範囲で分割数を 2 とすれば、正 12 角形の一部を作成できる。

また、4点 O, A_1, A_2, A_3 を多角形の線分に隠れないよう明示するため 08 行目で 4 点の大きさを大きくした。

09 行目の命令で 3 点 A_1, A_2, A_3 を結んだ閉曲線 sgCBDC を作成し、17 行目で、その内側の領域に斜線を引いた。

7.5 マクロの利用

次の場合分けの表は、増減表と同様に作ればよくて特に説明することがないように見えるかも知れない。

表 2 場合分けの表

	①	②	③	確率
2枚表 A	$0 < x \leq \frac{1}{3}$	$\frac{1}{3} < x \leq \frac{2}{3}$	$\frac{2}{3} < x \leq 1$	$\frac{1}{4}$
表・裏 B	$0 < f(x) \leq \frac{1}{6}$	$\frac{1}{6} < f(x) \leq \frac{1}{3}$	$\frac{1}{3} < f(x) \leq \frac{1}{2}$	$\frac{1}{2}$
2枚裏 C	$\frac{1}{2} < h(x) \leq \frac{2}{3}$	$\frac{2}{3} < h(x) \leq \frac{5}{6}$	$\frac{5}{6} < h(x) \leq 1$	$\frac{1}{4}$

実は、丸数字については、今回の報告で、ここまで紹介していないが、少し工夫が必要である。通常 $\text{T}_\text{E}_\text{X}$ は、コンパイル時に日本語変換で入力した丸数字など外字領域の文字は無視されてしまうので、○と数字を重ねたり、マクロ(パッケージ)を読み込んだりして対応する。自分が $\text{T}_\text{E}_\text{X}$ で文書を作成するときにプリアンブルで対応すればよいと思うかも知れないが、 $\text{K}_\text{E}_\text{T}_\text{C}_\text{i}_\text{n}_\text{d}_\text{y}$ で作成した図・表を確認するときには、一時的に作成された tex ファイルを利用する。一時的に作成されるファイルを見つけて必要なプリアンブルを加えることは、 $\text{K}_\text{E}_\text{T}_\text{C}_\text{i}_\text{n}_\text{d}_\text{y}$ の作業中にファイルが上書きされる上に、 $\text{K}_\text{E}_\text{T}_\text{C}_\text{i}_\text{n}_\text{d}_\text{y}$ のセキュリティ対策の面からも簡単ではない。対処方法は幾つかある。

- スクリプトの中に maru を定義する行を追加する。例えば、

```
TeXcom("\def\maru#1{\textcircled{\scriptsize #1}}");
```

という 1 行を加える。これにより、図の tex ファイルの中に

```
\def\maru#1{\textcircled{\scriptsize #1}}
}
```

という 1 行が加えられる。これにより、 $\backslash\text{maru}1$ として ① が使える。しかし、作成中の tex 文書から見れば、同じコマンドを 1 つの文書の中で繰り返し定義することがあり、気持ち悪い。

● 上と同じコマンドを定義して丸数字を使いたい場合、①, ②, ... を使うところに `maru1`, `maru2`, ... などを入れておいて、図の `tex` ファイルを利用するときに、「`maru` → `\maru`」という置換を実行しておく。これは、作図中の確認と出来上がりで頭を切り換える必要がある。また、置換によるミスも誘発する。

● 丸数字が使えるパッケージを使う。作成中の文書が丸数字が使えるパッケージを使っているなら、K_εT_Cindy での作業中に図・表を作成するための一時ファイルのプリアンブルに同じパッケージを加えることができればよい。1つ前に紹介した TeXcom コマンドは、図の `tex` ファイルに `tex` コードを加えるコマンドであるが、`Addpackage` は図・表を確認するための一時ファイルのプリアンブル部分にコードを加えるコマンドである。

スクリプトに

```
Addpackage("emath");
```

という 1 行を加えることにより、一時ファイルのプリアンブル部分に、

```
\usepackage{emath}
```

という 1 行が追加される。

8. 壁紙の利用

TeX で数学の問題リスト (pdf ファイル) を作ったのに、外枠とヘッダフッタを指定されたワードファイルを与えられたら、どうするだろう。TeX で同じように、外枠とヘッダフッタを用意するのは手間がかかりそうだとする。

1 つの対応は、それぞれ問題リストとワードファイルを印刷し、印刷物を切り貼りしたものをコピーして提出する方法である。

2 つめの対応は、ワードに pdf ファイルを貼り付ける方法である。ファイルとして提出するなら、ワードファイルになる。Mac OS 用のワードなら、pdf ファイルをオブジェクトとして選択し貼り付けるメニューがあるらしい。Windows 用のワードには、そのメニューが見つからないが、Acrobat Reader には、表示した pdf ファイルの必要な部分をコピーしてクリップボードに入れる機能があるので、クリップボードからワードに貼り付けることができる。

ここでは、第 3 の方法を紹介する。ワードファイルを TeX の壁紙として利用する方法である。その場合、ファイルとして提出するなら pdf ファイルになる。TeX で壁紙を利用パッケージもあるが、ここまで利用してきた K_εT_Pic で実現できる。なお、

K_εT_Cindy でなく、K_εT_Pic の `ketlayer` の話になる。

手順は、次の通り。

手順 1: ワードファイルを PDF ファイルとして出力する。それを例えば、「`wall.pdf`」としよう。

手順 2: `tex` ファイルの各ページの最初の部分に次の 3 行を加える。

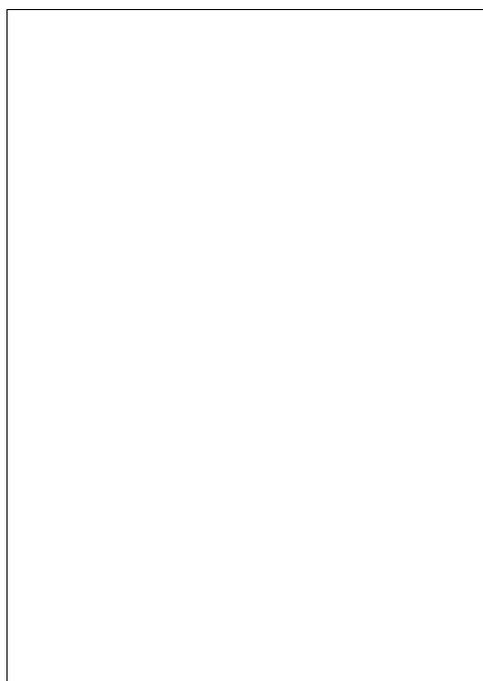
```
\begin{layer}{170}{0}
\putnотесе{-25}{-28.9}{\includegraphics
{wall.pdf}}
\end{layer}
```

`putnотесе` の数値は、目安としては、左と上の余白分だけ負の数にし、画像が用紙の左上の角からはじまるように調整する。

弓削商船の平成 29 年度編入学の公表用解答を例に紹介しておく。まず、ワードで指定された枠・ヘッダ・フッタは図 11 のようになっている。それを壁紙として、TeX で用意した公表用解答をコンパイルしたものが次のページの図 12 になる。

この方法は、どうだろうか。

(とじしろ)



1 / 1

平成 29 年度編入学者選抜 (数 学) M・1

図 11 用意された枠

(とじしろ)

問題番号	答 え	配点	問題番号	答 え	配点
問1	(1) $\frac{155}{30}$	2点	問5	(1) $\frac{1}{9}$	3点
	(2) $\frac{5x}{12}$	2点		(2) 3	3点
	(3) $-\frac{2a^2}{b}$ <small>※1</small>	2点		(3) 4	3点
	(4) $\frac{\sqrt{5}+1}{2}$	2点		(4) 6	3点
	(5) $6x^2+13x+6$	2点		(5) 5	3点
問2	(1) $a(3a-1)$	2点	(6) $-\frac{1}{2}$	3点	
	(2) $2(a+2)(a^2-2a+4)$	2点	(7) 0	3点	
	(3) $(3x+7y)(x-2y)$	2点	(8) $-\sqrt{3}$	3点	
問3	(1) $x=2$	3点	問6	(1) $3x^2-6$	4点
	(2) $x \leq -6$	3点		(2) $y=6x-7$	4点
	(3) $x = \frac{5 \pm \sqrt{13}}{2}$	3点	問7	(1) $\frac{1}{3}x^3 - x^2 - 3x + C$	4点
	(4) $3 \leq x \leq 5$	3点		(2) 6	4点
	(5) $x < -4$ または $x > 2$ <small>※2</small>	3点		問8	(1) $a=4$
問4	(1) $x=2$	3点	(2) $(-1, -4)$		4点
	(2) $x=27$	3点	(3) $\frac{1}{3}$		4点
	(3) $k=2$	3点	問9	(1) $\sqrt{2} \sin\left(x - \frac{\pi}{4}\right)$	4点
		(2) $x = \frac{5}{12}\pi$		4点	

注1. 「 $-2a^2b^{-1}$ 」なども正解とする。
 注2. 「 $x < -4, x > 2$ 」も正解とする。
 一方のみは不正解とする。

1 / 1

図 12 枠を壁紙として利用

9. 最後に

KeTcindy で作成した図・表を TeX 文書にレイアウトするには、前節で紹介した ketlayer が有効である。この原稿も ketlayer を使って図・表をレイアウトしている。図・表に限らず ketlayer は、文字列のレイアウトもできる。図・表の作成時に入れ忘れた式などがあっても、図・表に重ねてレイアウトすることができる。

KeTpic から始まる、効率的に正確な図を作成できる KeTcindy は、今も発展を続けており、マクロを利用する Addpackage も、2016 年 5 月に追加されたコマンドである。

もともと KeTpic は、TeX での図・表の利用を考えて開発されたものである。

不正確な図 (例えば補助線の間違い) が誤った結論を誘発することは、文献 [2] 第 4 章 (幾何学における間違い) にも紹介されている。特に (p.175)、正方形を幾つかの三角形に分け、長方形に組み替えると面積が変わってみえるという図は、有名だ。

KeTcindy のコマンド Figpdf により、作成した図を適切なサイズの pdf ファイルに出力することができる。図の出力サイズに自動で調整された pdf でありワープロなどに貼り込むときに、そのまま使えるので便利である。TeX に抵抗がある場合でも、

KeTcindy による正確な図の作成を試みてはどうだろうか。[url 4] は、KeTcindy での作図をより効率的にするためのヒントとして参考になると思う。

参考文献

- [1] CASTeX 応用研究会「KeTpic でらくらく TeX グラフ」, イーテキスト研究所, 2011.
 [2] A・S・ポザマンティエ, I・レーマン [著], 堀江太郎 [訳]「数学間違い大全集 誰もがみんなしくじっている!」, 化学同人, 2015.
 [3] 小山透 “TeX による学術専門書づくり”, 「数学通信」, 第 11 巻第 1 号, pp.62-73, 2006.

参考 URL

- [url 1] 「Ketpic.com」
 (<http://ketpic.com/>)
 [url 2] 「Scilab ユーザーグループ in Japan」
 (<http://scilab.na-inet.jp/>)
 [url 3] 「CinderellaJapan」
 (<https://sites.google.com/site/cinderellajapan/>)
 [url 4] 「KeTcindy による図入り教材の作成」
 (<http://www65.atwiki.jp/ketcindy/>)

ロゴについて

TeX 文書において、TeX というロゴは、\TeX というコマンドで出力する。このようなロゴが使えないときは、TeX (e が小文字) と表記することが共通認識となっている [3]。

それにならって、KeTpic の代用は KeTpic とし、KeTcindy の代用は KeTcindy と表記することが 2016 年 3 月末に、開発者の共通認識となった。