

KeTMath による課題送受・採点処理・結果分析

KeTCindy センター 高遠 節夫

Setsuo Takato, KeTCindy Center, Magnolia Inc.

長野高専・工学科 濱口 直樹

Naoki Hamaguchi, National Institute of Technology (KOSEN), Nagano College

山口大学・教育学部 北本 卓也

Takuya Kitamoto, Faculty of Education, Yamaguchi University

1 KeTCindy と KeTCindyJS

Cinderella は、1993 年から 1999 年にかけて、J.Richter-Gebert 氏と U.H.Kortenkamp 氏によって開発された動的幾何システムである ([1]). さらに、両氏は Cinderella に 3D 機能やプログラム言語 CindyScript を組み込んだ Cinderella.2 (以下 Cindy) をリリースした ([2]). CindyScript は通常の言語に近く、数値や文字列のデータ処理およびファイル操作が可能である. 一方、著者 (高遠) のグループは、2006 年以来、図のデータから Maple, Mathematica, Scilab, R により $\text{T}_\text{E}\text{X}$ の描画コード Tpic を書き出して、 $\text{T}_\text{E}\text{X}$ 文書の図作成に利用するためのマクロ集 $\text{K}_\text{E}\text{T}\text{pic}$ を開発してきた. この $\text{K}_\text{E}\text{T}\text{pic}$ の対話性をより高めるために、著者らは 2014 年に Kortenkamp 氏を招聘してジョイントワークショップを開き、その成果として $\text{K}_\text{E}\text{T}\text{Cindy}$ の最初のバージョン 1.0 ができあがった. $\text{K}_\text{E}\text{T}\text{Cindy}$ は以下の流れで $\text{T}_\text{E}\text{X}$ の図を作成する.

1. $\text{K}_\text{E}\text{T}\text{Cindy}$ の雛形ファイルをクリックして、Cindy を立ち上げる.
 - $\text{K}_\text{E}\text{T}\text{Cindy}$ のマクロライブラリ読み込みと実行ボタンの配置が実行される.
 - なお最近の Cindy では、Kortenkamp 氏に要請して $\text{K}_\text{E}\text{T}\text{Cindy}$ で用いられる Java 関数群を最初から `KetCindyPlugin.jar` として組み込んでもらっている.
2. Cindy の幾何要素 (点など) と $\text{K}_\text{E}\text{T}\text{Cindy}$ の描画関数から画面上に図を作成する.
 - 画面の図を見ながら、幾何要素や描画関数の引数を変更することができる.
 - 同時に描画コード作成のための R コマンドリスト (GLIST) を内部で作成する.
3. 画面上の Figure (描画ファイル作成ボタン) をクリックすると R が呼び出される.
 - まず、2 の GLIST により描画コードから成る $\text{T}_\text{E}\text{X}$ ファイルを作成する.
 - 次に $\text{T}_\text{E}\text{X}$ を起動して、確認用の $\text{T}_\text{E}\text{X}$ ファイルをコンパイル、PDF を作成する.
 注：図ファイルは `\input` で $\text{T}_\text{E}\text{X}$ ファイルに挿入される.

$\text{K}_\text{E}\text{T}\text{Cindy}$ からは、R だけでなく Maxima や gcc を呼び出すこともできる. gcc は $\text{K}_\text{E}\text{T}\text{Cindy}$ 曲面描画の隠線処理を高速化するためのものである. また、Maxima の数式処理計算の呼び出しは、例えば次のように行われる.

- 単関数の呼び出し

```
Mxfun("1","integrate",["x^2","x",0,1]);
```

注：`Mxfun("1","integrate("x^2,x,0,1",[]);`でもよい.

- 関数群の呼び出し¹

```
cmdL=[
  "i1:x^2-4*x", [],
  "i2:(1)/(x)+sin(x)", [],
  "o1:integrate([i1,i2],x)", [],
  "o1[1]::o1[2]", []
];
CalcbyM("ans",cmdL,[]);
```

- ・上記コマンドを ScriptEditor に記述して、スクリプトを実行すればよい。
- ・結果は `ans=["1/3*x^3-2*x","log(x)-cos(x)"]` となる。

2016年、ミュンヘン工科大学の Gebert 研究室グループは、Cindy とほぼ互換な HTML コンテンツを作成するフレームワーク CindyJS を開発して公表した²。これに対応して、Cindy にも CindyJS の HTML を書き出す機能が追加された。そこで、著者らは $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ で定義されている関数を組み込んだ HTML を作成する機能 $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{CindyJS}$ を $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ に追加した。 $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ の組み込みの手順は以下の通りである。

1. Cindy ファイル（例えば `sample.cdy` とする）のトップメニューにある「HTML に書き出す」を選択して CindyJS の HTML ファイル `sample.html` を出力する。
2. KeTJS のボタンを押すと、1 に $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ を組み込んだ HTML が作成される³。

上記の 2 を実行するプログラムについて少し説明を加える。まず、Cindy が出力するファイル `sample.html` を 1 行ずつ読み込む。そのためには、 $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ に追加されている関数 `Readlines` が用いられる。

```
Lines=Readlines(Dircdy,"sample.html");
```

ここで、`Dircdy` は Cindy ファイルのパスを返すグローバル変数で、やはり $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ で定義されている。次に、 $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ のライブラリを `sample.html` の Initialization スロット⁴に追加する。ただし、ライブラリの総行数は 20000 行を超えるので、ファイルサイズを抑えるために次の工夫を施している。

- $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ のライブラリで定義されているすべての関数について
関数名、先頭行、最終行、内部で使われている $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{Cindy}$ 関数群のリストをライブラリの更新ごとに作成する。
- `sample.html` の `draw`, `initialization` スロットで使われている関数を再帰的に拾い出して、その関数の定義だけを `initialization` スロットに追加する。

この方法により、総行数を 3000 行から多くとも 5000 行程度に抑えることができた。また、`sample.html` のファイルサイズは CindyJS と $\text{K}_{\text{a}}\text{T}_{\text{e}}\text{X}$ のライブラリを除き、ほとんどの場合 100KB 以下とかなり小さくなっている。「`ketcindy samples`」で検索されるサイト⁵には多くの $\text{K}_{\text{E}}\text{T}_{\text{C}}\text{CindyJS}$ のサンプルが掲載されている。

¹後述の KeT-LMS で実際に用いられる

²「CindyJS」で検索すると URL <https://cindyjs.org> が見つけられる

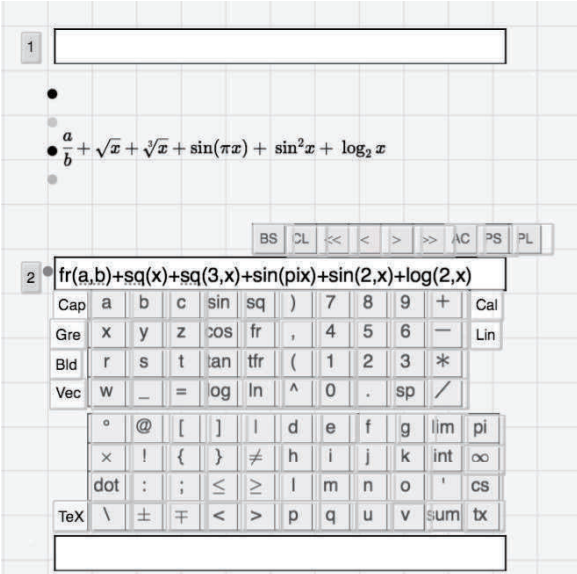
³On はホームページから `Cindy.js` を、Off はローカルフォルダから `Cindy.js` を読み込む

⁴Cindy ではプログラムの単位をスロットといい、`initialization` は初期設定の場合のみ実行される。

⁵<https://s-takato.github.io/ketcindysample/>

2 KeTMathの開発とKeTTaskによる問題の配付

2020年度はコロナ流行のため、多くの教育機関でオンライン授業の利用など、従来の授業方式の変更を余儀なくされた。著者の1人(高遠)が数学を担当している短期大学校でも、授業開始が6月に延期された上に、対面方式にせよ接触をできるだけ避ける工夫が必要となった。講義については、東邦大に所属するときからずっと $T_E X$ で作成したスライドを使用していた⁶ので、オンラインでも、双方向性を保ち授業の流れが単調にならないように工夫することで容易に対応できたが、プリントやPDFを用いない課題のやりとりでは数式をどうするかが問題になる。そこで、1次元の簡易数式ルールを作り、そのルールに従って記述された数式を即時に2次元 $T_E X$ 数式に変換して表示するHTMLアプリKeTMathを $K_E T C i n d y . J S$ で開発した⁷。例えば、分数 $\frac{a}{b}$ 、平方根 \sqrt{x} は、KeTMathルールではそれぞれ`fr(a,b)`、`sq(x)`と記述する。KeTMathの初期バージョンでは、PCやスマホのキーボードを用いてKeTMath数式を入力するようにしていたが、スマホのキーボードは種類が多いために、特殊記号を誤って入力してしまうという欠点があった。そこで、KeTMathの画面上にキーボードを配置するようにした。



The screenshot shows the KeTMath interface. At the top, there is a text input field containing the formula $\frac{a}{b} + \sqrt{x} + \sqrt[3]{x} + \sin(\pi x) + \sin^2 x + \log_2 x$. Below the input field is a keyboard overlay with various symbols and characters. To the right of the keyboard is a list of KeTMath rules, titled "KeTMath数式ルール".

KeTMath数式ルール

1. 分数 `fr(a,b)` 注) 小さい分数 `trf(a,b)`
2. 掛け算 `ab` 注) `a*b` も可
3. べき乗 `a^b(b)` 注) 1文字の場合は`a^b`も可
4. 平方根 `sq(a)`, 3乗根 `sq(3,a)`
5. 三角関数 `sin(x)`, `sin(x)`の2乗 `sin(2,x)`
6. 円周率 `pi`
7. 度 (`deg`)
8. 対数関数 `log(x)`, 自然対数 `ln(x)`, 底が`a` `log(a,x)`
9. 改行 `//`
10. スペース (`sp`) 注)\を出力
11. 立体 `100tx(m)`
12. 不定積分 `int(x^2,x)`, 定積分 `int(a,b,x^2,x)`
注) 定積分で関数の括弧を外すときは先頭に1を追加
13. ブラケット `br(f(x),a,b)`
14. 極限 `lim(x,a,f(x))` 注) 括弧を外すときは先頭に!
15. 和 `sum(k=1,n,k^2)` 注) 括弧を外すときは先頭に!
16. 微分 `diff(y,x)`, 偏微分 `par(z,x)`
17. 行列 `mat(a,b,c,d)`, 行列式 `det(a,b;c,d)`
18. 場合分け `case(a,(x<0);c;x(geq)0)`
19. ドット (`cdot`), \times (`cross`)
20. 複号 (`pm`), (`mp`)
21. 不等号 `<`, `>`, (`leq`), (`geq`), (`neq`)
22. 下添字 `a_n`
23. 全角文字を混せてもよい `2x-3=0`の解は`x=fr(3,2)`
24. ギリシャ文字 (`alpha`), (`beta`)
25. その他のTeX記号 (`sp`), コンマ, `{}`などで区切る

Fig.1 KeTMathの画面とKeTMathルール

KeTMathのプログラミングでは文字列の解析が重要である。Cindyの持つ関数`length`(長さ), `substring`(部分列), `tokenize`(分割), `indexof`(検索), `replace`(置換)に加えて、KeTMath数式を解析するための関数

`Bracket(文字列,"")` 括弧の位置とレベルを返す

`Getlevel(文字列,"","")` 括弧についてのコンマの位置とレベルを返す

などを`ketcindy`に組み込んでいる。

⁶スライドには適宜 $K_E T C i n d y$ で作成したフリップアニメーションを挿入している。

⁷「`ketcindy samples`」の他に「`ketcindy home`」にアクセスしてそのまま利用することができる。

また, KeTMath 数式を Cindy 形式に変換する Tocindyform, Maxima 形式に変換する Tomaxform も追加した. これらの関数を用いれば, 1つの KeTMath 数式から, TeX だけでなくグラフを描くための関数形式や Maxima で数式計算するための関数形式を出力することもできる.

Totexform("fr(pi,4)") $\implies \backslashdfrac{\{\pi\}}{4}$

Tocindyform("fr(pi,4)") $\implies (\pi)/(4)$

Tomaxform("fr(pi,4)") $\implies (\%pi)/{4}$

「高専・大学 数学・大日本図書」で検索される大日本図書の教科書ページ

https://www.dainippon-tosho.co.jp/college_math/

には上記の関数を利用して作成した Web Contents が掲載されている.

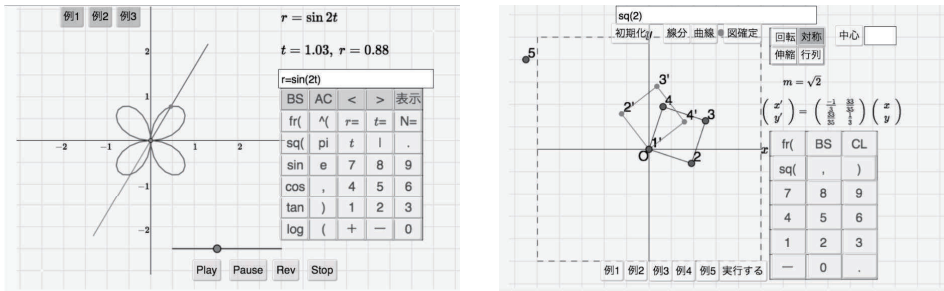


Fig.2 インタラクティブな Web 教材例

通常の授業では, 課題のやり取りはプリントで行われる. これに対して, KeTMath を単体で用いたときは次のようになる.

1. 学生が課題の書かれた PDF を見られるようにする. この PDF は講義スライドでもいいし, 独立に作成してもよい.
2. 学生はその課題をノートなどで解いて, 解答を KeTMath ルールで作成し, KeTMath で確認してから教員にメールなどで送信する.
3. 教員は回収した解答を KeTMath にコピー&ペーストして確認しながら採点する.

しかし, かなりの数の学生は KeTMath で確認することをせずに間違った数式をそのまま送ってきたり, $\sqrt{2}$ などと特殊文字を使ったりしていた. そこで, 問題自体を組み込んだ出題アプリ KeTTask を作成して配付することにした. 同時に, 教員が採点するためのアプリ KeTScore も作成することにして, それぞれの雛形ファイルから問題ごとのファイルを生成する K_{ET}Cindy のプログラム toolketmath.cdy を開発した. また, 問題の配付と回収はオンライン学習システム (以下 OLS) の1つである Google Classroom (以下 GC) を利用することにした⁸. まず, 課題の配付については次のようにする.

1. 1つのフォルダに kettaskorg.html, ketscoreorg.html, toolketmath.cdy⁹と CindyJS と KaTeX のライブラリフォルダ ketcindyjs を入れて, フォルダ data を作成する.

⁸個人ベースだったので GC を用いたが, Teams や Moodle などでも同様である.

⁹K_{ET}Cindy のライブラリを使うため, 「ketcindy home」からダウンロードしてインストールしておく.

2. dataの中に学生リスト student2022.txt と KeTMath ルールで記述された問題ファイル question1130-01.txt¹⁰を入れる。

学生リスト	問題ファイル
01AA	Q 注) ファイル名から番号追加
02BB	次の関数を微分せよ
03CC	[1] $y=x^4-3x^3+x^2+2x-3$
注) 番号+イニシャルなどの識別名	[2] $y=e^x+\log(x)$
	Sheet 注) 解答欄
	[1] $y'=::5$ 注) 配点
	[2] $y'=::5$
	Ans 注) 正解
	[1] $4x^3-9x^2+2x+2$
	[2] $e^x+\text{fr}(1,x)$

3. toolkitmath.cdy を立ち上げ、「taskline を作成」と「kettask に組み込み」のボタンを押すと、問題、解答欄、学生データを組み込んだ kettask.html ができる。
4. kettask.html ファイルを教員が用いているホスティング¹¹にアップして、リンク先の URL を取得する。
5. 4 の URL を OLS のテキストをやり取りする機能¹² を用いて学生に配付する。
6. 学生はスマホなどで kettask.html を立ち上げて、番号を入力確認してから KeTMath ルールで解答を入力する。
7. 解答が終わったら、Rec ボタンを押して最下段にある記録欄に表示される解答（1 行のテキストである）を OLS の回答にコピー＆ペーストして送信する。なお、解答の先頭には学生番号と提出日時のデータが追加される。

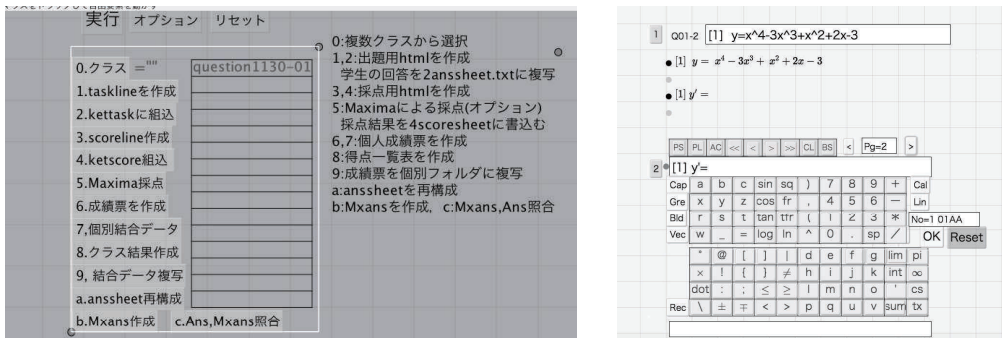


Fig.3 toolkitmath と kettask の画面

¹⁰2022 は年, 1130-01 は授業日 (回数) と問題番号. 以下のファイルでは省略.

¹¹著者らは github の Pages を利用している.

¹²GC, Teams, Moodle ではそれぞれ「質問」「クイズ」「アンケート」

3 KeTScoreによる採点

学生からの1行の回答は、OLSによって方法は多少異なるが、いずれを用いてもテキストの扱いは共通であり、1つのテキストファイル2anssheet.txt¹³に容易にまとめることができる。次に、toolketmath.cdyを立ち上げて、「scorelineを作成」と「ketscoreに組み込み」のボタンを押すと、問題、正解、配点、学生のデータを組み込んだ採点用のketscore.htmlができる。ketscore.htmlでは学生と問題がマトリックス形に配置されているので、学生ごとまたは問題ごとに採点することができる。画面の上段には正解、中段には学生の解答、下段には問題が表示され、それを見ながら中段の::の後に点数をつけていく。正解のときは、fullのボタンによってSheetに書かれた配点が入力される。採点が終わったらRecボタンを押すと、下段にすべての解答が1行のテキストとして表示されるので、これを確定データのためのファイル4scoresheet.txt¹⁴にコピー&ペーストすればよい¹⁵。採点をMaximaで行うには、toolketmath.cdyを立ち上げて「Maxima採点」のボタンを押すと、Maximaが起動して、問題ごとに得点を追加したファイル4_scoresheet.txtが生成される。証明問題のような場合は、Sheetの配点の後に::-1を追加しておくことMaximaでの採点が行われない。Maximaでの採点の結果は、scoresheetと同じ1行のテキストである。

1 01AA 111411:0:36 Q01—[1] $y' = 4x^3 - 9x^2 + 2x - 3$::5 [2] $y' = e^x + \text{fr}(1,x)$::5...
これをketscore.htmlの下段にコピー&ペーストすれば、KeTScoreでの採点と同様にし得点の確認修正をすることができる。

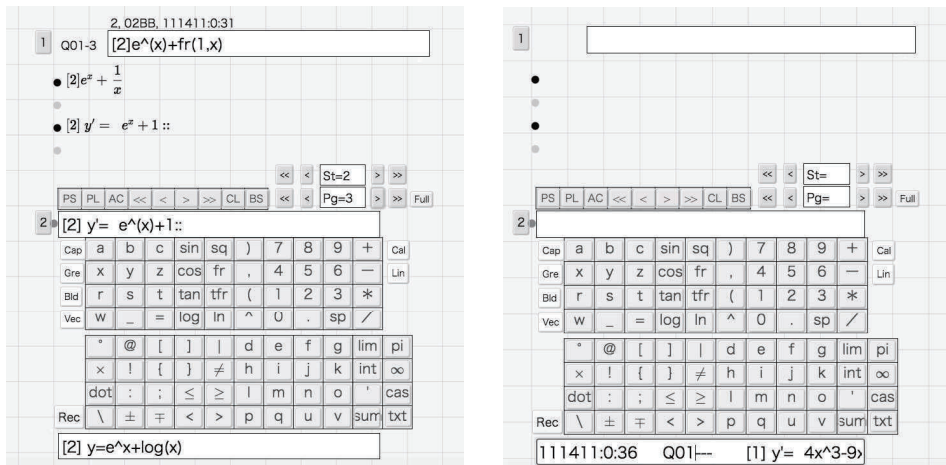


Fig.4 ketscoreの画面

KeTTaskを使う前は、解答をKeTMathに入力して確認してから送信するように指導はしたものの、スマホでは2つのアプリを同時に立ち上げて切り替えて実行することができないため、KeTMathを使わずにそのまま数式を送信してくる学生が多かった。その

¹³このファイルはtoolketmathで「tasklineを作成」を実行したときに準備される。

¹⁴scorelineを作成したときに準備される。

¹⁵中断して後で再開する場合もRecで表示されるデータをscoresheet.txtに保存しておく。

ため、Maximaでの採点時にエラーが頻発して実用的とは到底いえなかった。KeTTaskを用いるようになってからは、問題を解くときに即時に \TeX 数式が確認できるようになったので、入力ミスは激減して、Maximaでの採点の実用性が増している。

4 toolketmathによる結果の返却と分析

採点が終わって4scoresheet.txtが確定したら、toolketmath.cdyを立ち上げて「成績票の作成」と「個別結合データ」を順に押すと、学生ごと問題ファイルごとの個別成績票st01task1130-01.txt(st01は学生番号)と学生ごとに1つのテキストファイルにした成績票1130totalst01.txt(以下totalst.txt)がdata内のサブフォルダに作成される。

1130-01の結果

2 02BB 1130 11:0:31

Q01 次の関数を微分せよ。

[1] $y=x^4-3x^3+x^2+2x-3$

正解 $4x^3-9x^2+2x+2$

答え $4x^3-6x^2+2x+3$

得点 0

[2] $y=e^x+\log(x)$

正解 $e^x+\text{fr}(1,x)$

答え e^x+1

得点 0

.....

個々の学生の成績票totalst.txtには適宜コメントを追加することもできる。これを学生に送るために、著者らはDropboxのリンクを利用している¹⁶。具体的には

1. toolketmathのScriptにユーザーフォルダのパスDirdistを記述しておく。
Dirdist=Gethome()+"/Dropbox/" +成績フォルダ名;¹⁷
2. toolketmathの「結合データを複写」を押すと、Dirdist内の学生ごとのサブフォルダ(ない場合は新規に作成される)にtotalst.txtがコピーされる。
3. Dropboxで各フォルダのリンク先を取得して、学生に個別に知らせる。以後のデータも同じフォルダにコピーされるので、リンク先は1度だけ伝えておけばよい。
4. 学生はDropboxに登録していなくても3のリンク先にアクセスすることができる。

toolketmathの「クラス結果作成」のボタンを押すと、次のようなcsvファイルが作成される。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1				Q01					Q02					Q03				Q04		Q05				
2				[1]	[2]	[3]	[4]	[5]	[1]	[2]	[3]	[4]	[5]	[1]	[2]	[3]	[4]	[1]	[2]	[3]	[1]	[2]	[3]	
3	101AA	113010:09:47		5	0	5	5	0	5	5	5	0	3	0	0	0	5	5	5	5	3	5	5	66
4	202BB	113010:18:26		5	5	5	5	5	5	5	0	0	5	5	5	0	5	5	0	5	5	0	0	70
5	303CC	113010:11:34		5	5	0	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	95

¹⁶著者らは、他の方法に詳しくなく、Dropboxでのやり取りに習熟していることによる。

¹⁷Gethome()はKeTCindyの関数でユーザーホームのパスを返す。

5 Microsoft Teams を利用した授業実践

本節では、著者の一人（濱口）が Microsoft Teams を利用して行っている授業実践について報告する。実施している学年は高専の3年生で、今回は三角関数の微分公式の定着度を確認する以下の問題を設定した。

問題 1. $\sin x$, $\cos x$ の微分公式を書け。

問題 2. $\cos^2 x$ の意味は次のどれか。

(1) $\cos x^2$ (2) $(\cos x)^2$ (3) $(\cos)^2 x$

問題 3. 次の値を用いて $\cos^2 \frac{\pi}{3}$ の値を求めよ。

$\cos \frac{\pi}{3} = 0.5$ $\cos\left(\frac{\pi^2}{9}\right) = 0.46$ $\cos\left(\cos \frac{\pi}{3}\right) = 0.76$

問題 4. $\tan x$ の微分公式を書け。



Fig.5 Microsoft Teams の課題

学生に対しては、授業で Teams を利用するため PC やスマートフォンを準備しておくように連絡をしている。Teams では課題の「クイズ」を用いて、KeTTask のリンクとともに出题する。初めて用いる学生には、KeTTask の解答方法についての 5 分程度の説明を行うことになるが、KeTTask に、KeTMath ルールと解答方法を記載する方法も可能となっている。



Fig.6 課題設定までの画面

実際の授業では、手軽さもあり、対象学生のうちの 9 割以上がスマートフォンを利用している。今回の問題に対しては、出席した全員から授業時間内に提出されており、完了までに要した時間は平均しておよそ 15 分であった。

なお、Teams の「課題」を利用する場合、提出された解答のテキストデータの一覧は、学生の氏名等の情報とともに Excel ファイルで取得できる。



Fig.7 Teamsでの解答提出

Maximaによる採点を利用することにより、内容によってはクラス全体の各単位に関する理解度を授業時間内に確認することもできる。また、通常であれば翌週となることので多い解答返却も、Dropboxを利用することにより最短で当日中の返却が可能となっている。教員が授業内容の定着度を把握できるだけでなく、学生自身がすぐに理解度を確認できるこのシステムの利用により、教育手法の改善につながるものと考えられる。

今回の内容では、三角関数に関する数式記述の理解度についての確認も行っている。問題4の解答 $(\tan x)' = \frac{1}{\cos^2 x}$ の右辺について、正解の記述例としては、次の通りである。

- fr(1,cos(2,x)) (12名)
- fr(1,(cos(x))^2) (7名)
- 1/(cos(x))^2 (2名)

一方、不正解の記述例は以下の通りである。

- fr(1,cos^(2)) (xがない)
- fr(1,cosx^(2)) (xに括弧がない)
- fr(1.cos(x)^(2)) (コンマがピリオド)
- fr(1,cos^(2)(x)) (数式としては間違い)

また、これら以外にも fr(1,cos(x)^(2)) (4名)、1/cos(x)^(2) (1名) のように、ソフトウェアによっては正解となり得る記述例もあり、注意が必要である。

6 まとめと今後の課題

KeTMathを用いた課題のやりとり (KeT-LMSと呼ぶ) の手順は以下の通りである。

1. 課題.txt (問題, 解答欄, 正解) の作成 教員
2. 課題と解答のための kettask.html の作成 toolketmath
3. kettask.html の配付 OLSのプラットフォーム (教員)

4. 解答の作成と返送	学生
5. 解答の回収と一覧の作成	プラットフォーム (教員)
6. 採点用の ketscore.html の作成	toolketmath
7. Maxima を一部利用した採点	toolketmath
8. 学生個人の成績票作成	toolketmath
9. 成績票の配付	Dropbox リンクなど (toolketmath)
10. クラス成績表作成	toolketmath

上記のうち、1 (授業前) と 3,5 (準備は授業前, 配付と回収は授業前中) および 9 (成績票配付後の学生への通知) は教員の作業, 4 は学生の作業になるが, それ以外はすべて toolketmath のプログラムによって実行されるので, 教員は toolketmath のボタンを順に押すだけでよい。したがって, Maxima での採点がエラーなく実行されれば, 授業の途中でも課題の配付から返却までをスムーズに行うことができる。Maxima のエラーチェックは, 現在のところ

- KeTCindy の関数 Bracket を用いた括弧の整合性チェック
- \wedge の不正使用の有無 ($\sin(\wedge 2, x)$ など)

の2つであるが, 今後は通常授業での利用を増やして, エラーを同定し, 対応を KeT-LMS に組み込んでいきたい。

謝辞 本研究は JSPS 科研費 22K02972 の助成を受けている。

参考文献

- [1] Gebart J, Kortenkamp U., The Interactive Geometry Software Cinderella, Springer, 1999.
- [2] Gebart J, Kortenkamp U., The Cinderella.2 Manual, Springer, 2012.
- [3] Gagern M., Kortenkamp U., Gebart J., Strobel M., CindyJS- Mathematical Visualization on Modern Devices-, ICMS 2016, LNCS **9725**, 319–334, Springer, 2016.
- [4] 高遠節夫, 濱口直樹, Web 利用の理数教育に役立つ数式送受システムの開発, 京都大学数理解析研究所講究録 2178, 2021
- [5] 高遠節夫, 濱口直樹, 北本卓也, テキストをベースとした LMS の利用と HTML 教材の作成, 京都大学数理解析研究所講究録 2208, 2022
- [6] 高遠節夫, 濱口直樹, 北本卓也, 1 次元表現ルールに基づいた数式の送受と授業実践, 城西大学数学科数学教育紀要 (投稿中)