

数学的思考力と順序並び替え問題の自動生成

長坂耕作*

神戸大学 大学院 人間発達環境学研究所

(Received June 17, 2021 Revised September 27, 2021 Accepted November 21, 2021)

概 要

COVID-19 forced us to think of using several kinds of online learning environments where online quizzes and assessments are embedded and possibly auto-scored. In Mathematics, for so-called computational skills, we can prepare a set of auto-scored quizzes if the learning system can use with some CAS (computer algebra system). However, for mathematical thinking skills, it is quite difficult to prepare such quizzes especially for assessing any intermediate process of thinking. In this paper, we propose a way to drill and assess a part of mathematical thinking skills as a set of online quizzes on the Moodle with the ordering question type.

1 数学的思考力と数式処理を伴ったオンライン教育

中学校学習指導要領 [1] ならびに高等学校学習指導要領 [2] では、数学科の目標として「数学的に考える資質・能力¹⁾」を育成することが挙げられている。この「数学的に考える資質・能力」は、高等学校学習指導要領解説 [3] にも書かれているように、数学的に問題解決を行う「問題を見いだしたり、知識及び技能を活用して問題を解決したりする際に必要な思考力、判断力、表現力等」を求めていると解釈でき、中等数学教育のみならず、教養レベルの高等数学教育を含む幅広い数学教育で目指すべき共通する資質・能力であると言える。本稿では、特にこれら「思考力・判断力・表現力等」をオンライン教育で育成することについて、順序並び替え問題の自動生成を中心に、中等数学教育から高等数学教育まで幅広く対象として取り上げる。

1.1 「思考力・判断力・表現力等」とその教育方法

国立教育政策研究所の学習評価に関する参考資料 [5] において、思考力・判断力・表現力等の評価方法としては、一定の「考察が行われているか」を確認することが述べられている。そして、思考力・判断力・表現力等を伸ばす教材では、考察を実践させるために、試行錯誤と意見交換が中心的な役割を担っている。第 13 回数学教育国際会議の TSG(Topic Study Groups) 39 と 40 の

*nagasaka@main.h.kobe-u.ac.jp

¹⁾ 完全に同一のものであるかについては解釈の幅が若干あると考えられるが、本稿においては、英語での「mathematical thinking skills」に対応するものとして取り扱う(キー・コンピテンシーやスキル、資質・能力などの関係については、高口 [9] による研究報告書を参照されたし)。

報告 [22] でも, classroom assessment の基本デザインの 1 つとして, 解が複数あるもので, 解答それ自身よりも考え方を重視する教育的なモデルが挙げられている。

特に試行錯誤では, オープンエンドの課題 (オープンアプローチとも呼ばれ, 植村と高堂 [7] は, 正答がいく通りも可能になるように条件付けた問題と定義) が実践的に選ばれることが多い。例えば, Gibney[17] はアクションリサーチの中で, 算数教師を対象に, 数学的思考力を誘発する課題として Swan と Ridgway によるオープンエンド課題 [23] を取り上げている。うち 1 つを紹介すると, 様々な種類の階段 (平面図形) に対して「階段の急さ加減」を定義し議論するものである。教養レベルの微積分教育での数学的思考力においても, Breen と O'Shea[15] は, 授業が procedural fluency に特化しがちであることへの対応として, 連続性に関する指定した性質を満たす関数の例を挙げさせる課題 (原文では「non-routine questions」) などを取り上げている。

1.2 「思考力・判断力・表現力等」と自動採点

オンライン教育の特徴の一つである自動採点を不可欠な要素とするならば, 正答がいく通りも可能になるオープンエンドの課題ではなく, クローズドの課題を用いて, 思考力・判断力・表現力等を伸ばす必要がある。実際にどのような自動採点が数学で考えられるかについては, Hoogland と Tout[18] が低次の思考 (lower-order thinking) から高次の思考 (higher-order thinking) まで幅広く分類²⁾している。例えば, 全 7 段階の中央に位置する分類では, 「Sophisticated static contextual problems with depictive representations and interactivity in response space, but no interactivity in situation space」の問題に対して, 可能性のある自動採点方法として「Multiple Choice, Numerical field, Click on, drag and drop, pull down menu, matching, ordering, etc.」を挙げている。

この分類ないしはより高次の思考に分類される数学の自動採点可能な問題としては, Yerushalmy と Olsher[24] の動的幾何ソフト (GeoGebra) を使った例が挙げられるだろう。動的幾何ソフトはオープンエンドの課題での探求活動で使われることが多いが, Yerushalmy と Olsher の例では, 二者択一問題に付随する回答手段として使われている。高校生を対象とした与えられた初等幾何の命題の真偽を問う問題で, 例えば, 存在性を問い, 存在する場合には動的幾何ソフトを用いて具体例を作成させることで, 部分的な (あるいは全体を通して) 自動採点が可能となっている。

1.3 「思考力・判断力・表現力等」と数式処理, そして順序並び替え問題

数式処理は, 数学教育の様々な局面で活用可能であるが, 本稿においては, 自動採点可能な問題の自動作問を行うツールとしての側面にのみ着目する。特に自動作問には, Moodle の STACK のような統合された CAS を用いることで可能となる動的な作問 (これらのツールについては, Bokhove と Drijvers[14] が中等数学での代数処理を対象として比較を論じている) と, 著者による多肢選択問題の自動生成 [19] のような静的な問題としての作問が考えられるが, 本稿では後者の作問を取りあげる。つまり, あらかじめ生成しておくことで, 多くの環境で利用可能な可搬性・移植性を持つ静的な問題としての作問である。なお, 多肢選択問題は, Hoogland と Tout[18] による分類で, 低次の思考から中位程度の高次の思考に対応した分類に自動採点可能な回答方法として含まれているが, より高次の思考の分類 (上位 2 つの分類) には含まれていない。この違

²⁾ 論文自体は, 現在の数学教育を取り巻く CBAM(Computer-based assessment of mathematics) を取り上げている。

いは、問題と回答における双方向性の有無にある。多肢選択問題は、ある切り取られた状況下における選択を求めており、双方向性を確保することが著しく難しい問題形式となっている。

本稿では、高次の思考を必要とする問題形式であって、多肢選択問題に関しての先行研究 [20, 16] などでは有用とされている自習教材（定期的実施するドリル）に加える、新たな問題形式として、順序並び替え問題を提案する。順序並び替え問題は、史実を古い順に並び替えたり、化学反応の生じる準備に並び替えたりなど、知識を問う問題としての性質を思い浮かべられるかもしれないが、論理的思考力を問う国語科の教材としての研究も行われている。例えば、小野田ら [8] は、タブレット端末を用いて高校生を対象として、段落の役割を確認しながら文章を再構成する教材を実践しており、渡邊 [13] は、小学校低学年を対象に、文章と写真を照合しそれを正しい順序に並び替える活動を通して論理的思考力を高める試みを行っている。数学科に関する知見は少ないが、O'Connor らの調査 [21] によれば、小学校低学年の算数能力は、非数値的な並び替え能力と関係があることが判明している。また、高口 [9] は、先行研究に基づき、熟達者と初心者の違いが構造化された抱負な知識にあるということを示している。知識・技能を活用するための構造化には、全体を見通す俯瞰力が必要と考えられるが、実際に、中央教育審議会の 2040 年に向けた高等教育のグランドデザイン [4] では、俯瞰力が思考力・判断力・表現力と共に挙げられている。これらのことから本稿では、学習者が繰り返し利用可能な順序並び替え問題を提供することにより、知識・技能が構造化され思考力・判断力・表現力等が伸ばされることを期待し、数式処理を活用して大量の問題生成を行う方法を提案する。

以下、本稿では、実際に対象とする順序並び替え問題の自動採点可能な問題の形式を、Moodle 上の Ordering Question Type（この問題形式やそれを実現するプラグインに関しては、付録 A を参照のこと）とし、Moodle にインポート可能な大量な順序並び替え問題を生成するための枠組みを提供する Python 用のモジュールを第 2 章で紹介する。続く第 3 章では、数学分野において順序並び替え問題を作成する際のガイドラインを提案し、実際に生成した問題例を提示する。なお、本稿で想定している順序並び替え問題は、主に図 1 のような問題を解くための過程（または証明）を対象とした問題であり、既定の手順に並び替えるような知識を問う問題（前述の史実を古い順に並び替える問題など）ではないことに注意されたい。

2 Moodle XML Ordering Question Generator for Python

著者による多肢選択問題の自動生成 [19] を行うための Python 向けのモジュールである Moodle XML Question Generator をベースに、順序並び替え問題の生成を行えるように開発中のモジュールが「Moodle XML Ordering Question Generator」である。SymPy などの記号代数計算（数式処理）を組み合わせることにより、数学分野に特化した順序並び替え問題の生成が可能である。

2.1 モジュールによる順序並び替え問題の生成

まず、はじめに必要なのは、本モジュール `moodle_xog.core` の読み込みに加え、サポート関数を定義している `moodle_xog.qbank.common`、多くの問題生成に必要となる乱数生成用に `random`、そして数式処理のために `sympy` を読み込むことである。なお、以下の手順は全て Jupyter notebook 上で動作させることを想定していることに注意されたい。

次の計算を行う過程として適切な順に並び替えて下さい
(計算方法として最適なものとは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$
(問題整理番号:071)

→

次の計算を行う過程として適切な順に並び替えて下さい
(計算方法として最適なものとは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$
(問題整理番号:071)

図 1: 順序並び替え問題の例 (行列同士の積の計算)

```
import moodle_xog.core as mxog
import moodle_xog.qbank.common as mxog_common
import random
import sympy
```

このモジュールでは、問題生成器は、Question の小クラスとして定義する必要がある。定義の必要なメソッドは、question_generate と ordering_items_generate と question_text と answer_text の 4 つとなる。このうち、question_generate は実際に生成された問題インスタンスとして、Quiz クラスのインスタンスを出力する必要がある。インスタンス間の比較は、Quiz.quiz_identifier の比較で行うため、同じ問題であるかを判定可能なデータを格納しておく必要がある。以下の例では、数学的思考力という目的は脇に置き、整数を小さい順に並び替える問題生成器を取り上げている。

```

class integer_sorting_simple(mxog.Question):
    def __init__(self, number_of_items=4):
        self.num_of_items = number_of_items
    def question_generate(self, _quiz_number=0):
        quiz = mxog.Quiz(name=' 整数の並び替え', \
                           quiz_number=_quiz_number, lang='ja')
        # 問題データの生成 ( data:本体データ, quiz_identifier:問題比較用 )
        quiz.data = sorted(random.sample(range(-9,10), self.num_of_items))
        quiz.quiz_identifier = hash(''.join(map(str,quiz.data)))
        # フィードバック ( Moodle での回答後の表示に利用される )
        quiz.correct_feedback = ' 正解です。負や正の数の関係を理解しています。 '
        quiz.incorrect_feedback = ' 不正解です。符号の意味を再確認しましょう。 '
        return quiz

```

```

# 前ページ「class integer_sorting_simple(mxog.Question)」の定義の続き
def ordering_items_generate(self, quiz):
    # 正しい順序 ( order は 0 から順に増やす ) で選択肢 ( data ) を生成
    answers = [{ 'order':i, 'data':n} for i,n in
                zip(range(len(quiz.data)), quiz.data)]
    return answers
def question_text(self, quiz):
    return ' 次の整数を小さい順に並び替えよ。 <br />'
def answer_text(self, ans): # この場合は return ans['data'] でも動作
    return mxog_common.sympy_expr_to_text(ans['data'])

```

定義した小クラスを用いて問題生成器のインスタンスを作成後、モジュールの静的メソッドである generate を使用して、実際の問題を生成する。作成する問題数は、generate に size=100 のような形で指定する（無指定時は、10 個生成される）。

```

iss = integer_sorting_simple()
iss_quizzes = mxog.generate(iss, category=' 最小限の問題サンプル')

```

生成結果は、モジュールのクラス Quizzes のインスタンスとなり、listview メソッドで一覧表示を行うことや、preview メソッドでプレビュー表示が可能である（ただし、Moodle 上とは異なり、実際に並び替えることは出来ない）。最終的に、このようにして生成した結果の問題群は、save メソッドを使って Moodle にインポート可能な XML ファイルに保存できる。

```

iss_quizzes.save('integer_sorting_simple_in_japanese.xml')

```



図 2: 順序並び替え問題生成の例 (整数の並び替え)

実際に出力した XML ファイルを Moodle に読み込み、表示させたのが図 2 である。

2.2 複雑な問題の生成での留意点

このモジュールの問題生成器クラスが求めている 4 つのメソッドは、MVC モデル (Model-View-Controller に分割するソフトウェアアーキテクチャ) に倣っている。実際に問題生成を行う `generate` が Controller に対応しており、`question_generate` と `ordering_items_generate` が Model (個々の問題を数学的に確定させるデータや選択肢となる各項目データの生成処理)、`question_text` と `answer_text` が View (問題を実際に表示可能な状態にする処理) を担当する。この区別を行うことで、作成した問題生成器の再利用性が高まる。

実際に図 1 の問題生成器では、選択肢となる各項目データは次の構造を持っており、この構造に基づいて `answer_text` メソッドで、画一的な処理で最終的な HTML への変換を行っている。

- 選択肢 := 要素が「指示付き行列」であるか「 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文字列」である任意長のリスト
- 指示付き行列 := 「SymPy の行列」「要素の置換指示」「要素の修飾指示」のリスト
- 要素の置換指示 := 「置き換え対象の要素の位置」と置き換え先の「 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文字列」の集合
- 要素の修飾指示 := 「修飾する対象の要素の位置」と修飾に対応の「 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文字列」の集合

例えば、図 1 の左側の上から 2 番目の選択肢のデータは、概ね次のようになっている。

```
[[sympy.Matrix([[ -3]]), {}, {}], [sympy.Matrix([[ -4], [2]]), {}, {}], '+',
[sympy.Matrix([[5]]), {}, {}], [sympy.Matrix([[ -4], [3]]), {}, {}],
'\rightarrow', [sympy.Matrix([[ -8, -8], [9, 8]]), {(0, 1): ''}, (1, 1): ''},
{(0, 0): '\color{red}'}, (1, 0): '\color{red}']]
```

3 順序並び替え問題のガイドラインと生成例

本章では、前章で導入した順序並び替え問題の自動作問モジュールを活用するため、数学的思考力を伸ばす (定期的 to 実施するドリル向けの) 順序並び替え問題のガイドラインを提案し、それに基づく問題の生成について述べる。なお、本稿で扱う順序並び替え問題は、数学分野におけ

る「思考力・判断力・表現力等」を伸ばす問題を想定しており、直接的に知識・技能のみを確認するだけの問題は対象としていないことを改めて確認しておく。

本稿で重要と考えるのは、実際に「思考力・判断力・表現力等」を必要とする数学的な問題解決過程との関係性である。既に各校種の学習指導要領に基づき問題解決過程にかかる教育が行われており、それに基づく研究も進められている。例えば、谷地元ら [12] は、小学校算数科における問題解決的な学習とプログラミングとの関係性について考察を行っており、西村ら [10] は、数学的思考力を CBT(Computer Based Testing) で評価するための動的オブジェクトの数学的問題解決過程における役割について論じている。これらは伊藤の報告 [6] にあるような各種ある数学における問題解決モデルを基にしているが、ドリルのような自己学習として繰り返し行うことに直接的に反映するのは難しい。本稿では、適切なガイドラインに基づく順序並び替え問題が、数学的な問題解決能力の繰り返し学習としての可能性を持っていると考え、以下でそのためのガイドラインを提案する。

3.1 順序並び替えドリル問題のガイドライン

表 1 が提案する順序並び替えドリル問題のガイドラインであり、以下で、その理由を述べる。

3.1.1 明快

順序並び替え問題で問うべきものは、全体を見通す俯瞰力、関連する知識・技能と各項目の関係性、それらを組み合わせ与えられた問題を解決するための手続きとする論理的思考力などであり、難解な問題文を読み解く力とすべきではない。問題の読解力などを問うのであれば、多肢選択問題など別の問題形式を採用すべきであるが、多肢選択問題についても Zimmaro[25] により、「Ensure that the directions in the stem are clear, and that wording lets the examinee know exactly what is being asked」とのガイドラインが与えられていることに注意されたい(英文中の「stem」は選択肢ではない問題文のことを指す単語である)。

3.1.2 一意

自己学習の一環として実施されるドリルでは、教室内で行われるオープンエンドの課題に対するジグソー法や個別演習と異なり、個々の問題に対して十分なフォローアップをリアルタイムに行うことは難しい。そのため、正解と解釈可能な順序が一意に定まらない問題に教育効果は期待できない。例えば、図 1 の問題は正解の順序が一意になっているが、図 3 の左側のように一部を消去(一番上の項目の右側の行列の第 1 列を消去)すると一意でなくなる。この場合、図 1 の右

【明快】	問題文は求めている回答が明確で誤解が生じないよう単純明快にすること
【一意】	正解と解釈可能な並び替え順序は複数あってはならない。一意にすること
【均質】	並び替える各項目の粒度(各項目の論理的な大きさ)や表現は揃えること
【分離】	並び替える各項目に同じ式や文章を重複させてはならない。分離すること
【真摯】	紛らわしい項目は設定しない。ワイルドカードも最大で一つに抑えること

表 1: 提案する順序並び替えドリル問題のガイドライン

次の計算を行う過程として適切な順に並び替えて下さい
(計算方法として最適なものとは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$
(問題整理番号:071)

→

次の計算を行う過程として適切な順に並び替えて下さい
(計算方法として最適なものとは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$
(問題整理番号:071)

図 3: 一意でない不適切な順序並び替え問題の例 (行列同士の積の計算)

側の順序の他, 図 3 の右側の順序も正解と解釈できてしまう。また, 正解と解釈可能な全ての順序を列挙させる問題形式も考えられるが, 列挙させる必然性がない(知識・技能ではなく, 俯瞰力を含む思考力・判断力・表現力等を対象とした問題を議論している)。どうしても他の可能性を検討させたい場合は, 最適なものを提示した上で別解を回答させるのがよい。

3.1.3 均質

順序を並び替える各項目の粒度(正解の順序での全体像に対する, 各項目の論理的な大きさの割合)や表現が不揃いの場合, 各項目間の関係性や全体の俯瞰に加え, 特に大きな項目自体を独立して解釈する必要性が発生する。正解となる順序をフローチャート(プログラム)として考えると, この処理はサブルーチン呼び出す埋め込み構造(手続きの内容によっては, 再帰構造)に似ている。林[11]は, 大学生らを対象とした実験により埋め込み構造や再帰構造の場合の処理の困難さを示しており, 単なる埋め込み構造であってもワーキングメモリーの負荷が発生することを示唆している。実際の数学的な問題解決過程において, 補題などへの切り出し(埋め込み型構造)や数学的帰納法(再帰構造)が表れることは不可避と考えられるが, ドリルとして与えられた問題の解決過程(実際の事象ではなく, 順序を並び替える過程)には必ずしも必要とは言えない。そのため, 意図的に粒度や表現を揃えない場合を除き, これらは揃えることが望ましい。

3.1.4 分離

並び替えを行う各項目は, その表現において互いに分離独立している必要がある。分離独立されていない項目同士では, その関係性が自明となってしまう, 高次の思考を誘導することができない。例えば, 図 4 の左側の下側 2 つの項目を, 右側の下側 2 つのように書き直すと, 1 番目と 4 番目, 2 番目と 3 番目がそれぞれ分離されていない状態となり, 実質的に項目が 2 つの状態となってしまう。隣接する項目間の関係は, 正しい順序が一意となる必要性はあるが, それは深い考察により見出されるものにしなければならない(模範解答に表現等を揃える必要はない)。

次の計算を行う過程として適切な順に並び替えて下さい
(計算方法として最適なものは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$

(問題整理番号:071)

$(-2) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 4 \begin{pmatrix} -4 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} -8 & -8 \\ 9 & 8 \end{pmatrix}$

$(-3) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 5 \begin{pmatrix} -4 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} -8 & \\ 9 & \end{pmatrix}$

$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$

$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$

→

次の計算を行う過程として適切な順に並び替えて下さい (計算方法として最適なものは限りません)。

$$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix}$$

(問題整理番号:071)

$(-2) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 4 \begin{pmatrix} -4 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} -8 & -8 \\ 9 & 8 \end{pmatrix}$

$(-3) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 5 \begin{pmatrix} -4 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} -8 & \\ 9 & \end{pmatrix}$

$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix} \rightarrow (-3) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 5 \begin{pmatrix} -4 \\ 3 \end{pmatrix}$

$\begin{pmatrix} -4 & -4 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} -3 & -2 \\ 5 & 4 \end{pmatrix} \rightarrow (-2) \begin{pmatrix} -4 \\ 2 \end{pmatrix} + 4 \begin{pmatrix} -4 \\ 3 \end{pmatrix}$

図 4: 分離されていない順序並び替え問題の例 (行列同士の積の計算)

3.1.5 真撃

並び替えを行う項目の中に酷似したものが存在すると、本稿で意図する数学的思考力とは無関係に誤答を誘発する可能性がある。類似した項目の区別を評価したいのであれば、多肢選択問題などの他の問題形式を採用すればよいので、紛らわしい項目は設定してはならない。また、全ての項目を詳らかにするのではなく、敢えてワイルドカードのように見せない項目を設ける場合は、最大でも一つに抑えること(均質性とと同じく別の能力を求めることになってしまうため)。さらにこのとき、ワイルドカードの目的は、いたずらに問題を難解にするものではなく、本質的に決定できない場合に限定することが望ましい。例えば、可能性が複数考えられるが特定のものに限定することが適切でない場合や、その手順を明示することで項目間の分離性が失われてしまう場合などである。とにかく、自己学習のためのドリルであることに留意する。

3.2 ガイドラインに基づく順序並び替え問題の生成例

ガイドラインに基づく順序並び替え問題の例として、大学教養レベルの線形代数の授業で習う掃き出し法による線形方程式の求解の最後の段階を取りあげる。即ち、与えられた線形方程式に対応する拡大係数行列(または拡大行列とも呼ばれる)を構成し、その簡約な行列(または行簡約階段形ないしは行標準形とも呼ばれる)を行基本変形により計算した後に、結果からどのようにして解を構成するか、という問題である。ここでは、簡約な行列を連立方程式の形式に書き直さずに、行主成分などの性質を用いて解を書き下す方法の理解を促すことを目的とする。生成した結果が図5である(並び替え項目が8個あり、左側に詰める編集をした上で、その下に並び項目を右側に表示している関係で、実際のウェブブラウザでの表示状況とは多少異なる)。

この問題の自動生成は、次の手順で行っている。なお、第2章で紹介したモジュールの子クラスとして問題生成器を実装しているが、行列生成部分やHTML化などのViewに対応する部分は、別の問題生成器と共通の別クラスとして実装している。

1. 未知数の個数, 解の自由度, 制約の個数を指定された範囲内でランダムに定める。

ある線形方程式を解くために、拡大係数行列を簡約化したところ次の行列が得られました。この簡約行列から元の線形方程式の解を構成する過程として適切な順に並び替えて下さい。

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{array} \right)$$

(問題整理番号:032)

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

$$\vec{x} = \begin{pmatrix} \frac{1}{2} \\ 1 \end{pmatrix} c_1 + \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

$$\vec{x} = \begin{pmatrix} \frac{1}{2} \\ 1 \\ 0 \end{pmatrix} c_1 + \begin{pmatrix} 3 \\ 0 \\ -1 \end{pmatrix}$$

$$\vec{x} = \begin{pmatrix} \quad \\ \quad \end{pmatrix} c_1 + \begin{pmatrix} \quad \\ \quad \end{pmatrix}$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right) \rightarrow 3 - 2 = 1$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

$$\vec{x} = \begin{pmatrix} 1 \\ \quad \end{pmatrix} c_1 + \begin{pmatrix} 0 \\ \quad \end{pmatrix}$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

図 5: 掃き出し結果からの解の表現を導く問題の例

2. 係数行列の簡約な行列をランダム生成する（拡大係数行列ではない）。
3. 特殊解をランダムに定め、係数行列と積を求め、それにより拡大係数行列を生成する。
4. 並び替え項目を、想定する解の構成方法に倣い生成する。

この問題での数学的思考力に基づき解くまでの道筋の一例を述べる。

1. 解 \vec{x} の表現が 4 通り与えられており、表示されている数の多いものほど、順序は後ろになる。このことから、これら 4 つの項目の大まかな配置が推定できる。
2. 解 \vec{x} には任意定数 c_1 が使われており、解の自由度（未知数の個数と階数の差）を先に知る必要がある。これにより、最初の 2 項目が図 6 のようになることが確定できる。
3. 任意定数を 2 番目の未知数に導入することを決定づける理由として適切なものを、残りの項目から選択することになる。行主成分を含まない列に対応する未知数に任意定数を導入することになるため、これを示唆する 3 番目の項目が 4 番目の項目と共に確定できる。
4. 1 つ目の未知数に対応する一般解と特殊解がそれぞれ $\frac{1}{2}$ と 3 になることを決定づける理由として適切なものを、残りの 2 項目から選択することになる。1 列目に主成分を持つ行に対応するものなので、これを示唆する 5 番目の項目が 6 番目の項目と共に確定できる。
5. 最終的に残りの 7 番目の項目が 8 番目の項目と共に確定され、回答となる（図 6）。

このように順序並び替え問題は、知識・技能を活用可能であるかの評価に使えると共に、高次の思考を要求し、加えて、誤答時も正解を確認することで個々の操作の背景理解も促すことが可能である。自己学習のみならず、数学的思考力を評価する方法としても今後の可能性が期待される。

4 まとめと今後の課題

本稿では、数学的思考力（思考力・判断力・表現力等）を主眼に置きつつ、数学分野におけるオンライン教育での自習教材（定期的実施するドリル）の新たな問題形式として、知識以外を問

ある線形方程式を解くために、拡大係数行列を簡約化したところ次の行列が得られました。この簡約行列から元の線形方程式の解を構成する過程として適切な順に並び替えて下さい。

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{array} \right)$$

(問題整理番号:032)

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right) \rightarrow 3 - 2 = 1$$

$$\vec{x} = \begin{pmatrix} \\ \end{pmatrix} c_1 + \begin{pmatrix} \\ \end{pmatrix}$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

$$\vec{x} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} c_1 + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

$$\vec{x} = \begin{pmatrix} \frac{1}{2} \\ 1 \end{pmatrix} c_1 + \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

$$\left(\begin{array}{ccc|c} 1 & -\frac{1}{2} & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

$$\vec{x} = \begin{pmatrix} \frac{1}{2} \\ 1 \\ 0 \end{pmatrix} c_1 + \begin{pmatrix} 3 \\ 0 \\ -1 \end{pmatrix}$$

図 6: 掃き出し結果からの解の表現を導く問題の例 (各項目は正しい順に整列済み)

う順序並び替え問題を提案した。加えて、順序並び替え問題の作問に有用なガイドラインを作成すると共に、多肢選択問題を大量に生成するシステムを準用して、Moodle の Ordering Question Type 向けの問題を生成するシステムを開発した。以下では、本稿で紹介した線形代数分野の順序並び替え問題 (行列の積と線形方程式の解の表現) だけでなく、大学教養レベルの線形代数の授業で実際に自習教材として利用した経験を踏まえ、今後の課題について述べる。

4.1 順序並び替えドリル問題のガイドラインの課題

2021 年前期に学部 1 年次向け線形代数の授業で、実際に順序並び替え問題を大別すると 29 種類程度作成し、反転授業における教室外での事前・事後学修の一環として実施した (COVID-19 の影響により対面授業ではなく遠隔授業での実施となったため、教室内での演習等は Web 会議での実施)。自己評価となるが、本稿で提案しているガイドラインのうち、明快・一意・均質・真摯については、全ての問題が満たしていると考えるが、完全には分離されていない問題がかなりの割合含まれている。

この線形代数の授業では、行列の導入から掃き出し法による線形方程式の求解や逆行列の計算、行列式の計算までが範囲となっており、行の基本変形などの手続きをきちんと行えるかの計算問題が大きな割合を占める。このため、行列に表れる数値 (行列の成分) が選択肢間で時折多数共有されることが本質的に不可避である。また、反転授業の事前学修としては、適切な解法への気付きを誘導する必要性もあり、どの程度の分離性が適切であるかの判断が非常に難しい。数学の学習内容や学習過程のどの段階で課すドリル問題であるかによって、適切なバランスが存在するのかなども含め、今後の課題としたい。

4.2 数学的思考力育成に有効かの確認に関する課題

前述したように分離の適切性は難しい課題であるが、基本的には本稿で提案するガイドラインに沿って作成した問題を実際にドリルとして実施した。しかしながら、本稿で主眼を置いた数学

的思考力の育成に有効であったかの確認は行えていない。対象とした線形代数の授業内容では、思考力・判断力・表現力等を直接評価可能な定期試験等を実施することは難しいため、線形代数既学習者に対する聞き取り調査などが有効ではないかと考えており、中等数学教育での実践なども含め、今後の大きな課題の一つとして考えている。また、これらの課題に取り組む際には、作成することになる順序並び替え問題と、同じ単元(同じ学習過程)で用いられる記述式や多肢選択問題などの既存の問題との関係も明らかにすることで、より数学的思考力の育成に適した順序並び替え問題のガイドライン作成に繋がるのではないかと期待している。

謝 辞

埋め込み構造や再帰構造についてご教授くださいました林創先生、本稿の内容をより有意義にするご意見を頂いた査読者の方々に御礼を申し上げます。なお、本研究は JSPS 科研費 18K02941,21H00921 の助成を受けて行われたものを含んでいます。

参 考 文 献

- [1] 中学校学習指導要領(平成 29 年告示), 文部科学省, 2017.
- [2] 高等学校学習指導要領(平成 30 年告示), 文部科学省, 2018.
- [3] 高等学校学習指導要領(平成 30 年告示)解説 数学編 理数編, 文部科学省, 2018.
- [4] 2040 年に向けた高等教育のグランドデザイン(答申)(中教審第 211 号), 中央教育審議会, 2018.
- [5] 「指導と評価の一体化」のための学習評価に関する参考資料 中学校 数学, 国立教育政策研究所, 2020.
- [6] 伊藤俊彦: 発見学的戦略を用いた数学的問題解決過程の実験的研究について, 日本教科教育学会誌, 7(2), 1982, 55-63.
- [7] 植村哲郎, 高堂年幸: 数学教育におけるオープンエンドアプローチ, 鹿児島大学教育学部研究紀要. 人文・社会科学編, 36, 1984, 43-55.
- [8] 小田部明香, 小林祐紀, 田部成孝: 文章構成の意図の理解を目指すタブレット端末を利用した実践と評価, 日本デジタル教科書学会発表予稿集, 7, 2018, 77-78.
- [9] 高口努: 使って育てて 21 世紀を生き抜くための資質・能力, 資質・能力を育成する教育課程の在り方に関する研究報告書 1, 国立教育政策研究所, 2015.
- [10] 西村圭一, 安野史子, 根上生也, 祖慶良謙, 高橋広明, 浪川幸彦, 伊藤仁一, 三宅正武: 高大接続におけるタブレット端末利用の CBT の開発と検討: -数学問題における動的オブジェクトの役割-, 日本科学教育学会年会論文集, 41, 2017, 101-104.
- [11] 林創: 再帰的事象の認識とその発達に関する心理学的研究, 風間書房, 2008, 185 pages.
- [12] 谷地元直樹, 西條俊介, 三村仁: 算数科における「プログラミングの授業」のあり方に関する一考察: 問題解決的な学習とプログラミング的思考の関連に着目して, 北海道教育大学紀要 教育科学編, 71(2), 2021, 99-109.
- [13] 渡邊和幸: 説明的文章における, 論理的思考力を高める学習活動の工夫, 上越教育大学学校教育実践研究センター 教育実践研究, 21, 2011, 35-40.

- [14] Christian Bokhove, Paul Drijvers: Digital Tools for Algebra Education: Criteria and Evaluation, *Int J Comput Math Learning*, **15**, 2010, 45–62.
- [15] Sinead Breen, Ann O’Shea: The use of mathematical tasks to develop mathematical thinking skills in undergraduate calculus courses – a pilot study, *Proceedings of the British Society for Research into Learning Mathematics*, **31**(1), 2011, 43–48.
- [16] Mercedes Douglas, Juliette Wilson, Sean Ennis: Multiple-choice question tests: a convenient, flexible and effective learning tool? a case study, *Innovations in Education and Teaching International*, **49**(2), 2012, 111–121.
- [17] Janette Gibney: Provoking mathematical thinking: Experiences of doing realistic mathematics tasks with adult numeracy teachers, *Adults Learning Mathematics: An International Journal*, **9**(2), 2014, 97–115.
- [18] Kees Hoogland, Dave Tout: Computer-based assessment of mathematics into the twenty-first century: pressures and tensions, *ZDM Mathematics Education*, **50**(4), 2018, 675–686.
- [19] Kosaku Nagasaka: Multiple-choice questions in Mathematics: automatic generation, revisited, *Electronic Proceedings of the 25th Asian Technology Conference in Mathematics*, 2020, 21785-1 – 21785-15.
- [20] Wee Leng Ng, Kok Ming Teo, Khoon Yoong Wong, Kang Ling Kwan: The delivery role and assessment role of computer-based technology in a flipped university mathematics course, *Proceedings of the 24th Asian Technology Conference in Mathematics*, 2019, 185–194.
- [21] Patrick A O’Connor, Kinga Morsanyi, Teresa McCormack: Young children’s non-numerical ordering ability at the start of formal education longitudinally predicts their symbolic number skills and academic achievement in maths, *Developmental Science*, **21**(5), 2018, e12645.
- [22] Christine Suurtamm, Denisse R. Thompson, Rae Young Kim, Leonora Diaz Moreno, Nathalie Sayac, Stanislaw Schukajlow, Edward Silver, Stefan Ufer, Pauline Vos: Assessment in Mathematics Education: Large-Scale Assessment and Classroom Assessment, Springer International Publishing, 2016, 38 pages.
- [23] Malcolm Swan, Jim Ridgway: Classroom Assessment Techniques Mathematical Thinking, <http://www.flaguide.org/cat/math/math/math1.htm> (2021 年 6 月閲覧).
- [24] Michal Yerushalmy, Shai Olsher: Online assessment of students’ reasoning when solving example-eliciting tasks: using conjunction and disjunction to increase the power of examples, *ZDM Mathematics Education*, **52**(5), 2020, 1033–1049.
- [25] Dawn M. Zimmaro: Writing good multiple-choice exams, <https://facultyinnovate.utexas.edu/sites/default/files/writing-good-multiple-choice-exams-fic-120116.pdf> (2021 年 6 月閲覧).

A Ordering Question Type とその必要性

この付録では、数学分野の順序並び替え問題を Moodle で実現するために、Moodle に標準で含まれる問題形式でなく、https://moodle.org/plugins/qtype_ordering から別途インス

トールの必要がある Ordering Question Type を導入する必要性とその機能を簡単に紹介する。

Moodle に標準で含まれる問題形式には、順序並び替えを直接実現する問題形式はない。しかしながら、ドラッグ&ドロップイメージ、ドラッグ&ドロップテキスト、ドラッグ&ドロップマーカー、ミッシングワード選択などの問題形式を活用すれば、通常の順序並び替え問題は実現可能である（標準問題形式は、https://docs.moodle.org/en/Question_types を参照）。

それにもかかわらず、なぜ本稿では Ordering Question Type を導入しているのか。その理由は 2 つあり、どちらも数式の取り扱いに関する問題である。まず、ドラッグ&ドロップイメージなどの画像を用いる問題形式では、あらかじめ数式を画像化しておく必要がある。このとき、画像化自体に技術的な問題は存在しないが、受験者側（生徒・学生側）では問題が発生しうる。それは画面サイズの問題である。レスポンス画像を活用することで、通常の画像であれば画面サイズに適用した適切なサイズの画像ファイルを送付できるが、複数行に渡る数式では行送りの問題も発生するため、レスポンス画像での解決は難しい（もちろん、通信量の問題もある）。次に、ドラッグ&ドロップテキストなどのテキストを用いる問題形式では、残念ながら数式を扱うことができない。具体的には、マウスで操作する部分ではない問題文などでは、クライアントサイドの MathJax によるリアルタイム数式レンダリングが利用可能であるが、選択肢となる部分では利用できない。これらの背景から、マウスで操作する部分に $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ による数式を利用できる Ordering Question Type の導入を本稿では選択している。

なお、本稿で開発のモジュールでは積極的に活用していないが、Ordering Question Type では、選択肢の表示方法、提示する選択肢の選出方法、採点方法などが選べるようになっている。選択肢の表示方法は、Vertical/Horizontal からの選択となっており、数学分野でも短い数式を横に並べて表示することが適切な問題であれば、利用可能である。提示する選択肢の選出方法は、Select all items/Select a random subset of items/ Select a contiguous subset of items からの選択となっており、あらかじめ設定していた選択肢の中から、一部分のみを出題することを可能とする（例えば、手順 1,2,3,4,5 を準備しているが、実際に並び替えのために画面に提示されるのは、手順 2,4,5 のみなど）。数学分野の問題で活用することも不可能ではないが、本稿で求める数学的思考力ではなく、適切なのは知識を確認する問題と考えられる。採点方法は、All or nothing/Absolute position/Relative to correct position/ Relative to the next item (excluding last)/ Relative to the next item (including last)/ Relative to both the previous and next items/ Relative to ALL previous and next items/Longest ordered subset/Longest contiguous subset からの選択となっており、多様な評価が可能となっている。本稿のようにドリルとして活用するのではなく、成績評価として活用する場合は、Relative... や Longest... などの採点方法が有効と思われる。