

プログラミング初学者のためのパズル型問題開発と問題分析

西村 萌¹ 米澤 彩乃¹ 上島 綺夏¹ 黒木 奏子¹ 松澤 芳昭^{1,a)}

概要：本研究では、シャッフルされたプログラム断片を動作可能なプログラムに組み立て直す順序整列問題（パズル型プログラミング問題）の開発と評価をおこなった。コンピュータシステム上に実装されたパズル型問題は、操作履歴が獲得できるため、プログラム断片の操作という単位で思考過程の分析ができる。JavaScript, HTML, CSS を用いたプログラミング導入教育の受講者約 200 名に対して計 32 問の問題開発と実践を行い、解答の正誤分析と解答過程の分析を行った。正答率や解答過程での誤答選択肢の迷いより、プログラミング理解度が測定できるか考察を行った。本プログラミングテストの結果は、実プログラミングテストの結果と強い相関があり、プログラミング能力を測る問題として適切であるという示唆が得られた。S-P 表分析や共起マトリックス分析の結果、プログラミング初学者に適した問題は、出題ポイントが絞られており構造が明確な問題であること、不適切な問題は、難しい構造、間違いピースという 2 つの間違い要素を含んだ問題であることを考察した。出題の必要性が高いポイントとしては、文法事項や変数定義の位置について、必要性が低いポイントとしては、while 文の条件式の変数や乱数生成についてが挙げられた。

キーワード：プログラミング教育、プログラミング問題開発、ジグソー・コード、S-P 表分析、共起マトリックス

1. はじめに

プログラミング初学者がどのようにプログラム問題を思考するのか、という問いに際し、学習者が問題に解答する際の操作履歴の分析によりこれの解明を試みる研究が可能になってきた。本研究のテーマとする、シャッフルされたプログラム断片を動作可能なプログラムに組み立て直す順序整列問題（以下、パズル型プログラミング問題）でも操作記録を利用した思考プロセスの研究が行われてきた。しかし、現状ではまだプログラミング初学者に対して、どのような問題を出題することが適切または不適切か、プログラミングの理解度を適切に測ることができ問題とはどのような問題か、出題すべきポイントはどこかについては明らかではない。

そこで本研究では、パズル型プログラミング問題を 32 問開発し、プログラミング導入教育の受講者に対して出題した。その結果や解答パターン、解答過程を分析し、出題したパズル型プログラミング問題が適切であったか不適切であったか、またその要因や特徴、プログラムの理解度を適切に測ることができる問題はどのような問題かについて分析を行う。出題すべき必要性の高いポイントと低いポ

イントについての考察も行う。

2. 先行研究

本研究で扱うプログラム断片の順序整列問題は、Parsons ら [1] によって Parson's Problem と命名され、プログラミング教育研究の分野で一般的になっている。日本では、プログラム断片が紙を短冊上に分割した形をしていることから短冊形問題 [2] や、それを英訳した Split-Paper Testing [3] などと称されている。山口ら [4] は、ジグソーパズルを組み立てるようにプログラムを組み立てることからジグソー・コードと呼んでいる。Du らのレビュー論文 [5] によれば、15 年余りで、Parson's Problem 問題の分類、教育効果、プログラミング能力評価技法として妥当性、フィードバックの方法、支援ツールなどが議論されている。本論文では「パズル型問題」または「パズル型テスト」と呼ぶ。

Nakayama らは、Parson's 問題のプログラミング能力評価テストとしての妥当性を検証している [3]。Parson's 問題ではプログラム断片は与えられるのに対して、Constructed Response 型試験では被験者がスクラッチからプログラムを記述する実際のプログラミングによりを測定する。本論文ではこの方法を「実プログラミングテスト」と呼ぶ。Nakayama らの研究では 2013 年の情報科目の大学入学試験のパイロットテストでの Parson's 試験と実プログラミング試験を分析した結果、0.4 程度の弱い相関が得られた

¹ 青山学院大学 社会情報学部
School of Social Informatics, Aoyama Gakuin University
^{a)} matsuzawa@si.aoyama.ac.jp

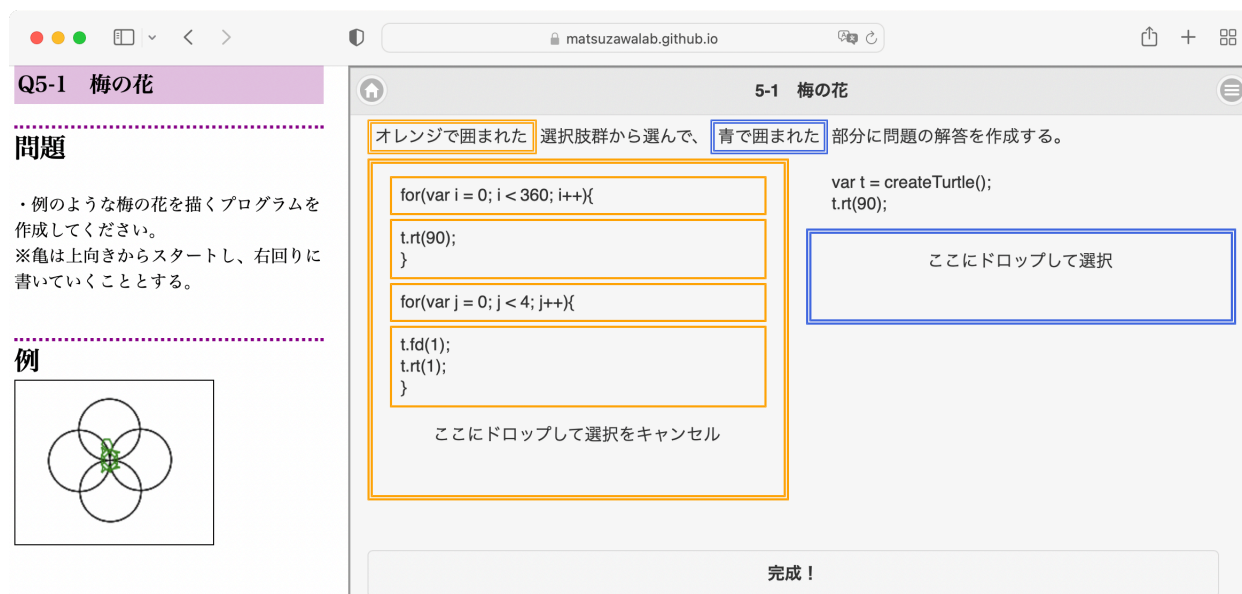


図 1 ジグソー・コードの画面例：5-1 梅の花

ことを報告している。Cheng らは [6] 完成されたプログラムをコード断片に分割し、誤り選択肢を加えることで Parson's 問題を生成する "The Code Mangler" を提案している。Cheng らの実験では、Parson's 問題の成績とプログラム記述問題の成績では 0.6 程度の相関が得られており、試験に係る時間は短くて済むことが示されている。

Parson's 問題とプログラム記述問題の本質的な差は、文法事項、またはアルゴリズムの各要素に選択肢が与えられていることである。これは、テキスト記述型プログラミングと Scratch に代表されるようなブロック型プログラミング環境の違いに類似していると考えられる。Zhi ら [7] は、ブロック型プログラミング環境 snap を利用して、ブロック型開発環境の中で Parson's 問題を提供するシステムを提案しており、ブロック型プログラミングの足場かけとなることが示されている。

元来 Parson's 問題は紙上で行われてきたが、近年では Web システムで行われることが一般的になっている。山口ら [4] は、Parson's 問題の開発・実験環境ジグソー・コードを提案している。コンピュータシステムでの試験を行うことの利点は、被験者の詳細な操作の記録、分析ができるようになることである。これにより、結果の分析だけでなく、プロセスの分析も可能にある。たとえば、ピースの共起行列と正答誤答パターンによるプロセス分析 [8] や、共起行列によって迷いを検出できる可能性の検討 [9]、ピースの操作状態遷移の類型分析 [2] などが試行されている。

3. パズル型問題の開発

3.1 研究の目的

本研究の目的は、パズル型プログラミング問題が学習者のプログラムの理解度や思考過程を測ることができるかを

確認することである。加えて、初学者に出題するのに適切な問題の特徴を明らかにすることである。本研究のリサーチクエストは、以下の二つである。

RQ1 プログラミング初学者に出題する問題として適しているまたは不適切なのはどのような問題かを明らかにすること

RQ2 解答パターンや解答過程の分析によって出題すべきポイントを見つけることができるかを明らかにすること

3.2 ジグソー・コード

ジグソー・コードは、山口らが開発を進めているパズル型プログラミング問題の実験システムである [4]。今回、我々はジグソー・コードを用いて、オリジナルの問題を開発した。

ジグソー・コードの画面例を図 1 に示す。解答を開始すると、画面左側に課題プログラムの仕様、右側にオレンジ色枠と青色枠が提示される。課題プログラムの仕様では問題文と出力結果が表示される。出力結果は、タートルプログラムであれば出力される図形、計算プログラムであれば出力仕様といったように問題に合わせて表示される。オレンジ色枠には必要ピースと不必要ピースが表示され、青色枠が解答欄である。まず、解答者は課題プログラムの仕様通りになるピースを探す。ピースをドラッグ&ドロップで解答欄に移動して適当な順序に並び替える。

3.3 問題開発

本研究で開発する問題の対象者は、青山学院大学社会情報学部社会情報学科 1 年生の必修授業「コンピューティング実習」の受講者である。パズル型プログラミング問題は、

表 1 パズル型テスト問題一覧

番号	単元	問題名	正答率	注意係数	合 計 ピース	不 要 ピース	文法	アルゴ	ジャン ル
2-1	変数	正三角形	0.40	0.49	14	6	20%	80%	G
2-2		適正体重	0.38	0.51	8	4	40%	60%	T
2-3	変数	矢印	0.67	0.52	4	0	0%	100%	G
2-4		釣り銭	0.58	0.50	8	4	80%	20%	T
3-1	条件分岐	図形書き分け	0.36	0.50	13	6	60%	40%	G
3-2		今日傘持って行く？	0.48	0.55	12	4	70%	30%	T
3-3		くじ引き	0.49	0.38	8	3	60%	40%	T
3-4	条件分岐	偶数？奇数？	0.07	0.25	11	5	60%	40%	T
4-1	繰り返し	一等星	0.58	0.55	6	2	60%	40%	G
4-4		MITSUBISHI マーク	0.32	0.62	6	3	20%	80%	G
5-1	入れ子構造	梅の花	0.73	0.27	4	0	0%	100%	G
5-2		3 が出た回数	0.12	0.37	7	2	30%	70%	T
5-4		星を 6 個描く	0.52	0.50	8	4	20%	80%	G
6-2	関数	四角と三角を交互に並べる	0.29	0.57	8	2	60%	40%	G
6-3		7star	0.79	0.45	4	2	50%	50%	G
6-4		感染状況を求める	0.21	0.42	9	2	20%	80%	T
6-5		ドーナツ	0.24	0.45	8	4	20%	80%	G
7-1		関数 (戻り値つき)	0.66	0.34	6	3	80%	20%	T
7-2		ラッキーナンバー	0.61	0.31	6	2	20%	80%	T
7-3		手裏剣	0.09	0.21	10	2	20%	80%	G
8-1	HTML	index.html の作成	0.66	0.51	10	3	-	-	-
8-2		next.html の作成	0.73	0.44	9	0	-	-	-
9-1	CSS	index.css の作成	0.51	0.56	4	1	-	-	-
9-2		table.css の作成	0.40	0.49	6	3	-	-	-
11-1	HTML と Javascript の連携	入場料	0.45	0.33	5	1	80%	20%	T
11-2		誕生月占い	0.47	0.24	5	1	0%	100%	T
11-3		31 がある月をリスト表示	0.52	0.57	5	2	80%	20%	T
12-1	Canvas 入門	白黒ボックス	0.31	0.34	4	0	0%	100%	G
12-2		四角形を表示する	0.44	0.60	4	0	0%	100%	G
12-3		ラッキーナンバー 2	0.58	0.37	5	3	80%	20%	T
13-1	タイマーとアニメーション	おむすびころりん	0.35	0.37	4	0	0%	100%	G
13-4		カメ	0.38	0.36	5	0	0%	100%	G

授業で各週の冒頭で行われる小テストで実施するために開発された。小テストは前週までの範囲を理解しているかを確認するために行われた。

パズル型プログラミング問題を 32 問開発した。その一覧を表 1 に示す。表 1 中、「番号」は問題番号、「単元」は各週で扱った授業内容、「問題名」は各問の問題の名前、「正答率」は各問題の正答率、「注意係数」は各問題の注意係数、「合計ピース」は必要ピースと不必要ピースの合計、「不要ピース数」は不必要ピースの総数、「文法」と「アルゴ」は文法とアルゴリズムの比率、「ジャンル」は各問題の種類を示している。

「注意係数」とは S-P 表分析で得られる指標で、その問題に対する異質度を示し 0~1 の値をとる。注意係数が低い場合は、問題の難易度に対して正解すべき受験者が正解し、不正解すべき受験者が不正解する問題である傾向であることを示す。注意係数が高い場合は、正解すべき受験者が間違い、不正解すべき受験者が正解してしまう傾向があることを示す。

問題の特徴を踏まえて問題を分析するために、「文法」

「アルゴリズム」と「問題ジャンル」指標を各問題に付与した。「文法」と「アルゴリズム」については、各問題の誤りポイントがどの程度の割合を占めるかを示したもので、3.4 節「誤りポイント」で詳説する。「問題ジャンル」は、タートルグラフィックスのように画像表示を伴うプログラムの問題を G、テキスト出力をするプログラムの問題を T と分類し、表記する。

3.4 誤りポイント—文法とアルゴリズム

著者らは、多くの学生が間違えると想定した要素を誤りポイントとした。

誤りポイントの例を図 2 を利用して説明する。この問題における必要ピースは太枠内にある s1, s2, s3, s4, s5, s6 の計 6 個、不必要ピースは s7, s8, s9, s10, s11 の計 5 個である。誤りポイントは、

AP01 s1 と s7, s8 のいずれかを選択させることで、指定した範囲の乱数を生成できるかを問う。アルゴリズムの理解度を図ることができる。

AP02 s2 と s9 のどちらかを選択するさせることで、割

s1	var number = random(40) + 10;
s2	var even = number % 2;
s3	if(even != 0){
s4	println('奇数です。'); }
s5	else {
s6	println('偶数です。'); }
s7	var number = random(50) + 10;
s8	var number = random(10) + 49;
s9	var even = number / 2;
s10	else if {
s11	if(even != 0){

図 2 問題例：3-4 偶数？奇数？

り算の演算子を理解しているかを問う。文法の理解度を図ることができる。

AP03 s5 と s10 のどちらかを選択させることで、比較演算子を理解しているかを問う。文法の理解度を図ることができる。

の 3 つである。

想定した誤りポイントが 3 つあり、うち 2 つが文法要素を、残りの 1 つがアルゴリズム要素である。さらに条件分岐の構造を組み立てる仕様になっているため、文法の比率は 60%，アルゴリズムの比率は 40%と設定した。このように、文法、アルゴリズムの比率は、誤りポイントを列挙し、割合を算出した上で出題者の感覚で多少の調整を加えることで算出している。

3.5 評価実験

評価実験は 2021 年度の「コンピューティング実習」で第 3 週から第 14 週まで、計 12 回行われた。被験者数は 207 名であった。そのうち、完全データが得られた 201 名を分析の対象とした。パズル問題は 1 問あたり 2 分から 4 分、計 6 分から 8 分で行った。^{*1}

3.6 分析方法

問題分析では 3 つの方法を用いた。1 つ目は S-P 表分析 [10] である。S-P 表で著者らが作成した問題を分析する。これによって S-P 表が完成し、注意係数を用いて問題の傾向を分析する。著者らの開発した問題が典型パターンのどれに当てはまるか、「良問」、「難問」、「要検討」のいずれに該当するか、その問題の特徴と傾向を分析する。

2 つ目は、実プログラミングテストとパズル型プログラミング問題との相関関係についての分析である。実プログラミングテストは、指定されたプログラムを作成するという問題を 80 分間で 4 問解くという内容で、パズル型プログラミング問題とは別の問題を出題した。パズル型プログラミング問題が実力を測るものとして適しているか、その

^{*1} 被験者がパズル型テストに慣れるまでは 1 問あたり 4 分で行い、慣れてきた段階で 2 分と設定した。

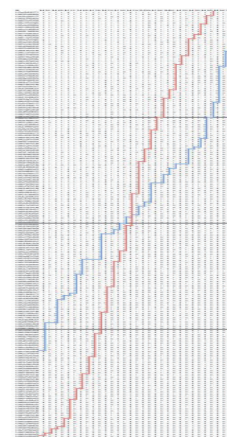


図 3 S-P 表の形状

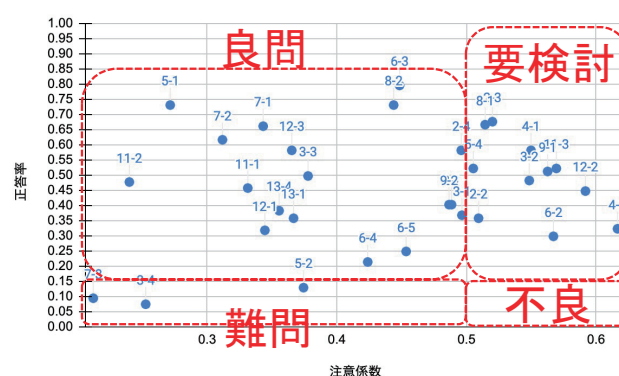


図 4 注意係数の分布

妥当性についての分析を目的とする。

3 つ目は山口らの提案する共起マトリックス [4] である。共起マトリックスを使い、「良問」、「難問」、「要検討」と分類された問題の思考過程を分析する。

4. 結果

4.1 S-P 表分析

4.1.1 S-P 表の形状

パズル型プログラミング問題の解答結果より作成した S-P 表を図 3 に示す。図 3 より、我々が開発した問題の傾向として、問題の正答率が 20% から 80% に分布していること、得点率が 50% ほどの生徒が最も多いこと、0 点や満点に近づくにつれて生徒数の分布が少なくなっているということが挙げられる。ここで、S-P 曲線の形状分析を行った。形状の種類を表 2 に示す。その結果は「テスト型 (I)」であり、これは、標準学力テストや実力テストといった多数の生徒の学力を総合的に測る場合にみられる分布である。この結果より、我々が開発した問題は、学生のプログラミング能力を的確に測ることが出来ていたといえる。つまり、実力測定に適している問題であったと考えられる。

4.1.2 問題の分類

各問の正答率と注意係数との関連を図 4 に示す。S-P 表

表 2 S-P 曲線の代表的な形 (文献 [10] より作成)

形状	概要
テスト型 (I)	点数の分布が正規分布型になっている。標準学力テストや実力テストなどで見受けられる形状。
テスト型 (II)	P 曲線が斜めに直線的かつ S 曲線が斜めに直線的。達成度テストで見受けられる形状。
演習問題型	一部の少数の問題の達成度が他の問題に比べて低い。大多数の生徒は平均点前後の達成度に集中しているが、一部の下位の生徒の達成度が突然低くなっている。
ドリル型	ごく少数の生徒の達成度がやや低め。ドリルによくみられる形状。
両極端型	クラス全体の達成度が高い問題群と低い問題群とが両極端に分断されている。好ましくない形状。
プログラム学習型	スキナー型のプログラム学習でみられる形状。
ブリテスト型	全問題にわたってクラスの達成度が低く、生徒の達成度はごく少数の生徒を除外し非常に低い。ブリテストで見られる形状

分析 [10] に倣い、以下の基準で「良問」、「難問」、「要検討」、「不良」の 4 グループに分類した。

良問 正答率 0.15~0.85, 注意係数 0.3~0.5

難問 正答率 0.15 以下, 注意係数 0.5 以下

要検討 正答率 0.15~0.85, 注意係数 0.5~0.6

不良 正答率 0.15 以下, 注意係数 0.5~0.6

開発した問題において、良問に分類される問題は全体の約 6 割、難問に分類される問題は全体の約 1 割、要検討に分類される問題は全体の約 3 割であった。不良に該当するものはなかった。該当したグループの特徴を以下に示す。

良問とは、受講者がプログラミングのレベルに応じて正解することが出来ている問題である。良問に分類される問題の特徴は

- ・ 誤りポイントにおいて間違えている人の割合が高いこと
- ・ 誤りポイントが文法要素かアルゴリズム要素かのどちらかに偏っている場合が多いこと

が挙げられる。つまり、誤りポイントが明確である方が、正しくプログラミング能力を測定することができると考えられる。

難問とは、良問と同様に受講者がレベルに応じて正解することが出来ているが、正答率が非常に低い問題である。難問に分類される問題の特徴は、

- ・ 関数や繰り返しに組み込まれる条件分岐が問われている確率が高いこと
- ・ 構造が難しいかつ文法的な不必要ピースがあることが挙げられる。つまり、アルゴリズムが複雑な問題は初学者にとって難易度が高く、正答率が低くなってしまうと考えられる。

要検討とは、受講者がレベルに応じた解答が出来ていない可能性がある問題である。要検討に分類される問題の特徴は、

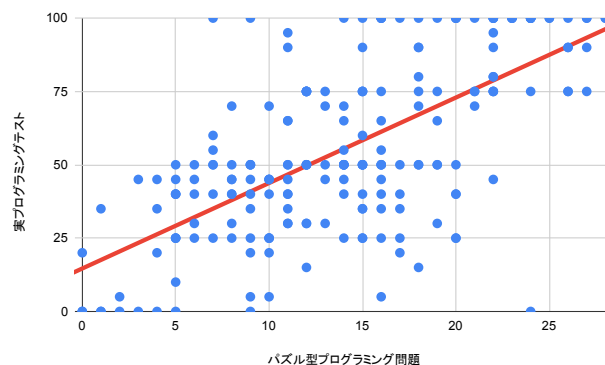


図 5 パズル型プログラミング問題と実プログラミングテストとの相関

- ・ 誤りポイント以外の場所で間違いが見られること
- ・ アルゴリズムはわかっているが、文法で間違える人が一定数いたこと

が挙げられる。つまり、誤りポイント以外で間違える要因があったため、意図した部分の理解度の測定が難しい傾向にあると考えられる。

画像表示を伴う問題とテキスト出力を主とする問題で問題の分類に差が出るかについても検討したが、関連性はほとんどみられなかった。実装することができないパズル型テストでは、画像表示を伴う問題は頭の中で図形をイメージすることが難しく、要検討になりやすいと考えていた。実際は、問題ジャンルが分類に影響を及ぼすことはないと考えられる。

4.2 実プログラミングテストとパズル型プログラミング問題の相関

パズル型プログラミング問題の得点と実プログラミングテストの得点の相関分析を行った。散布図を図 5 に示す。スピアマンの順位相関係数は、 $\rho=0.632(p<.01)$ であった。この結果より、今回開発した問題と実プログラミングテストとの間には、正の強い相関があることが分かった。Nakayama ら [3] による、実プログラムテストとパズル型プログラミング問題の得点の相関は、 $\rho=0.478$ であることを考慮すると、今回開発した問題の相関は非常に強いといえる。

パズル型プログラミング問題は、実プログラミングテストと違いプログラムを実装することができない。このような違いがあるにも関わらず、相関が強いのは、実プログラミングテストを解くことができない人は、パズル型テストのようにピースがあっても、正しく並び替えることができないからだといえる。

4.3 共起マトリックス分析

パズル型プログラミング問題を正答率と注意係数から、良題、要検討、難問の 3 つの種別に分類した。各種別の共

問題	結果	正解者	不正解者																																																																																																																																																																																																																																																																																																																		
<pre>s1 var number = random(40) + 10; s2 var even = number % 2; s3 if(even != 0){ s4 println('奇数です。'); s5 else { s6 println('偶数です。'); s7 var number = random(50) + 10; s8 var number = random(10) + 49; s9 var even = number / 2; s10 else if { s11 if(even != 0){</pre>	<table><tr><td>正答率</td><td>7%</td></tr><tr><td>注意係数</td><td>0.25</td></tr></table> <table><tr><td>解答パターン</td><td></td></tr><tr><td>ピースの並び順</td><td>人数 割合</td></tr><tr><td>s1, s2, s3, s6, s5, s4</td><td>25 12</td></tr><tr><td>s1, s2, s11, s6, s5, s4</td><td>24 12</td></tr><tr><td>s1, s2, s3, s4, s5, s6</td><td>15 7</td></tr><tr><td>s1, s9, s11, s6, s5, s4</td><td>11 5</td></tr><tr><td>s1, s9, s3, s6, s5, s4</td><td>8 4</td></tr><tr><td>以下省略</td><td></td></tr></table>	正答率	7%	注意係数	0.25	解答パターン		ピースの並び順	人数 割合	s1, s2, s3, s6, s5, s4	25 12	s1, s2, s11, s6, s5, s4	24 12	s1, s2, s3, s4, s5, s6	15 7	s1, s9, s11, s6, s5, s4	11 5	s1, s9, s3, s6, s5, s4	8 4	以下省略		<table><tr><td>s1</td><td>s2</td><td>s3</td><td>s4</td><td>s5</td><td>s6</td><td>s7</td><td>s8</td><td>s9</td><td>s10</td><td>s11</td></tr><tr><td>s1</td><td>1</td><td>6</td><td>2</td><td>1</td><td>1</td><td>0</td><td>1</td><td>2</td><td>2</td><td>0</td><td>2</td></tr><tr><td>s2</td><td>0</td><td>5</td><td>6</td><td>2</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>4</td></tr><tr><td>s3</td><td>1</td><td>0</td><td>1</td><td>7</td><td>1</td><td>4</td><td>0</td><td>0</td><td>2</td><td>0</td><td>1</td></tr><tr><td>s4</td><td>1</td><td>0</td><td>0</td><td>5</td><td>7</td><td>7</td><td>0</td><td>1</td><td>1</td><td>3</td><td>0</td></tr><tr><td>s5</td><td>0</td><td>1</td><td>0</td><td>3</td><td>1</td><td>9</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>s6</td><td>0</td><td>1</td><td>0</td><td>5</td><td>3</td><td>0</td><td>0</td><td>1</td><td>2</td><td>3</td><td>2</td></tr><tr><td>s7</td><td>7</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>4</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>s8</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>3</td><td>0</td><td>0</td><td>0</td></tr><tr><td>s9</td><td>0</td><td>5</td><td>2</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>3</td><td>0</td><td>1</td></tr><tr><td>s10</td><td>0</td><td>0</td><td>1</td><td>0</td><td>4</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>s11</td><td>0</td><td>0</td><td>6</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>5</td></tr></table>	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s1	1	6	2	1	1	0	1	2	2	0	2	s2	0	5	6	2	0	1	1	0	0	0	4	s3	1	0	1	7	1	4	0	0	2	0	1	s4	1	0	0	5	7	7	0	1	1	3	0	s5	0	1	0	3	1	9	0	0	0	1	0	s6	0	1	0	5	3	0	0	1	2	3	2	s7	7	1	1	0	0	0	4	1	0	0	0	s8	2	1	0	0	0	0	1	3	0	0	0	s9	0	5	2	0	1	0	1	0	3	0	1	s10	0	0	1	0	4	2	0	0	0	0	0	s11	0	0	6	2	0	0	0	0	2	0	5	<table><tr><td>s1</td><td>s2</td><td>s3</td><td>s4</td><td>s5</td><td>s6</td><td>s7</td><td>s8</td><td>s9</td><td>s10</td><td>s11</td></tr><tr><td>s1</td><td>22</td><td>31</td><td>9</td><td>2</td><td>6</td><td>8</td><td>19</td><td>15</td><td>30</td><td>11</td><td>11</td></tr><tr><td>s2</td><td>5</td><td>30</td><td>26</td><td>17</td><td>11</td><td>21</td><td>6</td><td>4</td><td>19</td><td>10</td><td>36</td></tr><tr><td>s3</td><td>3</td><td>10</td><td>34</td><td>5</td><td>11</td><td>33</td><td>3</td><td>4</td><td>24</td><td>10</td><td>31</td></tr><tr><td>s4</td><td>13</td><td>10</td><td>14</td><td>30</td><td>17</td><td>28</td><td>10</td><td>2</td><td>8</td><td>18</td><td>11</td></tr><tr><td>s5</td><td>0</td><td>12</td><td>9</td><td>70</td><td>20</td><td>13</td><td>1</td><td>6</td><td>12</td><td>21</td><td>9</td></tr><tr><td>s6</td><td>4</td><td>3</td><td>14</td><td>54</td><td>70</td><td>30</td><td>3</td><td>3</td><td>11</td><td>22</td><td>9</td></tr><tr><td>s7</td><td>29</td><td>10</td><td>6</td><td>6</td><td>5</td><td>9</td><td>18</td><td>16</td><td>18</td><td>6</td><td>2</td></tr><tr><td>s8</td><td>18</td><td>14</td><td>2</td><td>3</td><td>5</td><td>6</td><td>17</td><td>24</td><td>10</td><td>3</td><td>5</td></tr><tr><td>s9</td><td>2</td><td>49</td><td>28</td><td>3</td><td>4</td><td>23</td><td>2</td><td>1</td><td>34</td><td>6</td><td>23</td></tr><tr><td>s10</td><td>5</td><td>11</td><td>13</td><td>24</td><td>28</td><td>10</td><td>1</td><td>6</td><td>7</td><td>23</td><td>5</td></tr><tr><td>s11</td><td>3</td><td>9</td><td>30</td><td>12</td><td>12</td><td>39</td><td>5</td><td>1</td><td>9</td><td>13</td><td>34</td></tr></table>	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s1	22	31	9	2	6	8	19	15	30	11	11	s2	5	30	26	17	11	21	6	4	19	10	36	s3	3	10	34	5	11	33	3	4	24	10	31	s4	13	10	14	30	17	28	10	2	8	18	11	s5	0	12	9	70	20	13	1	6	12	21	9	s6	4	3	14	54	70	30	3	3	11	22	9	s7	29	10	6	6	5	9	18	16	18	6	2	s8	18	14	2	3	5	6	17	24	10	3	5	s9	2	49	28	3	4	23	2	1	34	6	23	s10	5	11	13	24	28	10	1	6	7	23	5	s11	3	9	30	12	12	39	5	1	9	13	34
正答率	7%																																																																																																																																																																																																																																																																																																																				
注意係数	0.25																																																																																																																																																																																																																																																																																																																				
解答パターン																																																																																																																																																																																																																																																																																																																					
ピースの並び順	人数 割合																																																																																																																																																																																																																																																																																																																				
s1, s2, s3, s6, s5, s4	25 12																																																																																																																																																																																																																																																																																																																				
s1, s2, s11, s6, s5, s4	24 12																																																																																																																																																																																																																																																																																																																				
s1, s2, s3, s4, s5, s6	15 7																																																																																																																																																																																																																																																																																																																				
s1, s9, s11, s6, s5, s4	11 5																																																																																																																																																																																																																																																																																																																				
s1, s9, s3, s6, s5, s4	8 4																																																																																																																																																																																																																																																																																																																				
以下省略																																																																																																																																																																																																																																																																																																																					
s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11																																																																																																																																																																																																																																																																																																											
s1	1	6	2	1	1	0	1	2	2	0	2																																																																																																																																																																																																																																																																																																										
s2	0	5	6	2	0	1	1	0	0	0	4																																																																																																																																																																																																																																																																																																										
s3	1	0	1	7	1	4	0	0	2	0	1																																																																																																																																																																																																																																																																																																										
s4	1	0	0	5	7	7	0	1	1	3	0																																																																																																																																																																																																																																																																																																										
s5	0	1	0	3	1	9	0	0	0	1	0																																																																																																																																																																																																																																																																																																										
s6	0	1	0	5	3	0	0	1	2	3	2																																																																																																																																																																																																																																																																																																										
s7	7	1	1	0	0	0	4	1	0	0	0																																																																																																																																																																																																																																																																																																										
s8	2	1	0	0	0	0	1	3	0	0	0																																																																																																																																																																																																																																																																																																										
s9	0	5	2	0	1	0	1	0	3	0	1																																																																																																																																																																																																																																																																																																										
s10	0	0	1	0	4	2	0	0	0	0	0																																																																																																																																																																																																																																																																																																										
s11	0	0	6	2	0	0	0	0	2	0	5																																																																																																																																																																																																																																																																																																										
s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11																																																																																																																																																																																																																																																																																																											
s1	22	31	9	2	6	8	19	15	30	11	11																																																																																																																																																																																																																																																																																																										
s2	5	30	26	17	11	21	6	4	19	10	36																																																																																																																																																																																																																																																																																																										
s3	3	10	34	5	11	33	3	4	24	10	31																																																																																																																																																																																																																																																																																																										
s4	13	10	14	30	17	28	10	2	8	18	11																																																																																																																																																																																																																																																																																																										
s5	0	12	9	70	20	13	1	6	12	21	9																																																																																																																																																																																																																																																																																																										
s6	4	3	14	54	70	30	3	3	11	22	9																																																																																																																																																																																																																																																																																																										
s7	29	10	6	6	5	9	18	16	18	6	2																																																																																																																																																																																																																																																																																																										
s8	18	14	2	3	5	6	17	24	10	3	5																																																																																																																																																																																																																																																																																																										
s9	2	49	28	3	4	23	2	1	34	6	23																																																																																																																																																																																																																																																																																																										
s10	5	11	13	24	28	10	1	6	7	23	5																																																																																																																																																																																																																																																																																																										
s11	3	9	30	12	12	39	5	1	9	13	34																																																																																																																																																																																																																																																																																																										
				(太枠内が正解選択肢)																																																																																																																																																																																																																																																																																																																	

図 6 3-4 「偶数?奇数?」問題と分析結果

問題

```
s2 for(var j = 0; j < 4;j++){
s3 for(var i = 0; i < 360; i++){
s4 t.fd(1); t.rt(1); }
s5 t.rt(90); }
```

結果

正答率	73%
注意係数	0.27

解答パターン

ピースの並び順	人数	割合
s2, s3, s4, s5	147	76
s2, s5, s3, s4	8	4
s3, s4	7	3
以下省略		

正解者

	s2	s3	s4	s5
s2	26	80	26	54
s3	27	6	125	15
s4	42	21	28	92
s5	19	16	4	22

不正解者

	s2	s3	s4	s5
s2	8	15	13	18
s3	12	4	25	15
s4	15	10	10	8
s5	7	4	8	12

(太枠内が正解選択肢)

図 7 5-1 「梅の花」問題と分析結果

起マトリックスから読み取ることができる思考過程について、それぞれの例を1つずつ取り上げ、結果を以下に示す。

4.3.1 「3-4 偶数?奇数?」

問題、解答パターン、正解者および不正解者の共起マトリックスを図6に示す。この問題は乱数で指定された範囲の数を生成し、偶数か奇数かを判断することによって、条件分岐で出力結果を変える問題である。出題意図は、乱数の生成と条件分岐、演算子の文法問うことである。不必要ピースは5つである。そのうち2つは発生させる乱数の範囲の間違いが含まれ、残りの3つはそれぞれ四則演算子、if文、比較演算子の文法間違いが含まれる。

この問題は正答率は7%、注意係数が0.25であり難問に分類される。正答率は非常に低いが注意係数が低く得意な人や理解ができていない人は解くことができる問題である。したがって問題としては適切だったと考えられる。

解答パターンでは、誤りポイント以外で間違えている受験者はほとんどいなかった。不正解者はs6とs4を入れる場所を逆にしている人が最も多い。if文の分岐で出力させる結果を逆にしていることから、条件分岐のアルゴリズムを正しく考えられていないと推測できる。2番目に多いのは必要ピースs3の代わりに不必要ピースs11を入れてしまうという間違いである。したがって演算子「!=(ノットイコール)」の文法を覚えていない人が多いと考えられる。

共起マトリックスより、正解者は上から順にピース列を整える傾向が強いことが読み取れる。s7からs1、s11からs3への共起が強いことから、不必要ピースの間違いに気づき修正することができている。

不正解者は不必要ピースであるs9からs2への共起が強い。不必要ピースであるs9の誤りに気づき修正していることから、四則演算子に関しては理解ができている者が多いと考えられる。

正解者不正解者に共通して、必要ピース同士の共起と不必要ピースから必要ピースへの共起の両方が見られる傾向にある。したがってこの問題は誤りポイントの通り、アルゴリズムと文法の両方を問う問題であった。

4.3.2 「5-1 梅の花」

問題、解答パターン、正解者および不正解者の共起マトリックスを図7に示す。この問題は繰り返し文を入れ子構造にすることで図形を描く問題である。出題意図は入れ子構造を理解しているか問うことである。不必要ピースがないアルゴリズム問題である。

この問題は正答率は73%、注意係数は0.27であり、良問に分類される。したがって問題としては適切だったと考えられる。

解答パターンでは、誤りポイント以外で間違えている受験者はほとんどいなかった。不正解者はs5とs3の順序を逆にしている人が最も多い。このことから不正解者はs3

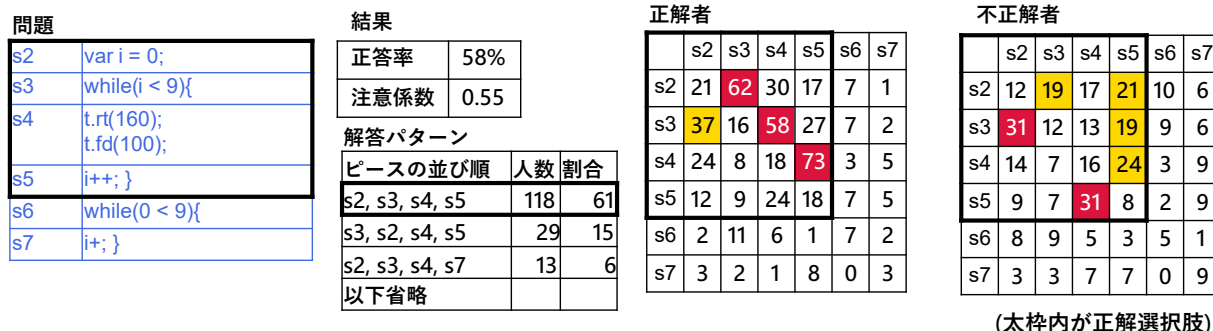


図 8 4-1 「一等星」問題と分析結果

と s4 を組み合わせて円を描き、s2 と s5 を組み合わせて 4 回繰り返すという繰り返し文は理解できているが、繰り返しの中に繰り返しを入れるという入れ子の構造を理解できていないと考えられる。

共起マトリックスより、正解者は上から順にピース列を整える傾向が強いことが読み取れる。s2 からの共起が最も強いのは s3 で、続いて多いのが s5 である。ここから正解者は入れ子構造の外側の繰り返しから先に考えピースを並べている人が多いと考えられる。

不正解者は不必要ピースである s9 から s2 への共起が強い。不必要ピースである s9 の誤りに不正解者は s2 から s5 と、s3 から s4 の共起が強い。このことから不正解者は 2 つの繰り返し文をそれぞれ分けて考えている傾向がある。

4.3.3 「4-1 一等星」

問題、解答パターン、正解者および不正解者の共起マトリックスを図 8 に示す。この問題は繰り返しを使って図形を書く問題である。不必要ピースは 2 つあり、それぞれ while 文の条件式の間違いが含まれるピースと、演算子の間違いが含まれるピースである。

この問題は正答率は 58%、注意係数は 0.55 であった。したがって要検討に分類される問題である。

解答パターンより不正解者は s2 と s3 の位置を逆にしていう者が最も多かった。これは誤りポイントにはなかったが、不正解者の多くが while 文の変数を定義する位置を理解できていないと考えられる。次に多かったのは誤りポイントの一つである s7 を選択している者である。s7 は演算子に関する文法の間違いが含まれるピースであることから、演算子の理解が曖昧であると考えられる。

共起マトリックスより、正解者は上から順にピース列を整える傾向が強いことが読み取れる。正解者は s3 から s2 の共起がみられる。このことから s3 の変数使用が含まれるピースを先に並べてから定義する場所を考えている者が一定数いると考えられる。

不正解者は s2, s3 から s5 への共起と、s5 から s4 の共起が強い。このことから不正解者の中には正解者と違い上からピースを並べずに、繰り返しの構造を作ってから中身を

考えている者がいると考えられる。

正解者と不正解者に共通して、必要ピース同士の共起が強いという特徴がある。つまり文法よりもアルゴリズムを問う問題であった。誤りポイントは文法であり予測と違う結果になったことから再検討する余地があると考えられる。

5. 考察

5.1 RQ1. 「プログラミング初学者に出題する問題として適しているまたは不適切なのはどのような問題かを明らかにすること」

プログラミング導入教育の受講者 201 名に対して今回出題した問題は、S-P 表分析によって、「テスト型 (I)」に分類されることが分かった。実力テストや標準学力テストなどで多く見受けられる形である。相関分析より、実プログラミングテストとパズル型プログラミング問題の結果に相関がみられた。したがってパズル型プログラミング問題はプログラミングの実力測定に有用であったと考えられる。

本プログラミングテスト 32 問は、良問 18 問、要検討 11 問、難問 3 問に分類することができた。

良問は、誤りポイントが文法要素かアルゴリズム要素かのどちらか偏っている場合が多い。このような誤りポイントが明確な問題では、構造を理解しやすくケアレスミスでの間違いが起ころういため、プログラミング能力が高い人と低い人を的確に測ることができる可能性が高いからだと考える。

要検討は、アルゴリズムよりも文法で間違えている者が多く、類似したピースが多い。文法問題に関しては、プログラミング能力が高い人も低い人も曖昧な場合が多いこと、たまたま正解してしまうこともあるからだと考えられる。

そして、難問は、アルゴリズムが難しいかつ文法的な間違いピースも、正答者でも良問に比べて、上から順番にピースを並べ替えていないという特徴があった。これは、難しいアルゴリズム、間違いピースという 2 つの間違い要素があるため、正答者でも迷いが生じやすいからだと考えられる。

5.2 RQ 2. 「解答パターンや解答過程の分析によって出題すべきポイントを見つけることができるかを明らかにすること」

共起マトリックスを利用して解答過程や迷いの分析を行った結果、誤りポイントでの間違いや迷いが生じている問題は 32 問中 18 問であった。このような問題には、 $=$ と $==$ の違いや $!=$ 、変数定義、`innerHTML` と `value` の違いなど、文法に関する誤りポイントで間違えている学生が多い傾向がある。多くの学生が、文法事項について曖昧であることから、初学者に覚えてもらうためのポイントとしては適切であると考えられる。

反対に、誤りポイントで間違いや迷いが生じていない、または想定していない間違いが多い問題は 32 問中 14 問だった。条件式の中の変数が数値になっている誤りポイントで間違える人は少なく、変数定義をする位置を間違えている人が想定より多かった。よって、`while` 文の条件文の変数部分についてや乱数生成の際に足す数については誤りポイントにする必要性が低く、変数の位置を問う部分は誤りポイントとする必要性が高いと考えられる。

5.3 本研究の限界

本研究の共起マトリックスを用いた解答過程や迷いの分析では、文法事項によって生じている解答過程での迷いは、ピースを工夫することによって分析することができた。しかし、アルゴリズムに関する部分の迷いについては、ピース数が増えるほどに間違いパターンや解答者がピースを動かす回数が増えてしまう傾向にあるため、正確に分析することが難しい。パズル型プログラミング問題で、アルゴリズムの解答過程や迷いを分析するためには、幅広い間違いパターンの検討と共起マトリックスの多数派ではないピースの動きについても分析する必要があるだろう。

6. まとめ

本研究では、パズル型プログラミング問題の開発、S-P 表分析や共起マトリックスを用いた問題の分析を行った。良問に分類される問題の特徴として、出題意図が明確で構造が組み立てやすい問題である、誤りポイントの通りに間違えている人が多いという特徴があった。また、共起マトリックスの解答パターンや解答過程の分析から考えることができる出題すべきポイントとしては、文法に関する部分や、変数定義をする位置が挙げられた。

しかし、初学者の理解度を測るのに適したピース総数や間違いピースの数、ピースないの文量、文法とアルゴリズムの比率については、明確ではない。そのため、今後の展望としては、そのような点を明らかにすることで、プログラミング初学者の理解しづらいポイントやつまづきやすいポイントを的確に把握し、学習環境の改善に結びつけることができるかを明らかにしたいと考えている。

参考文献

- [1] Parsons, D. and Haden, P.: Parsons' Programming Puzzles: A Fun and Effective Learning Tool for First Programming Courses, *Proc. 8th Australasian Conference on Computing Education (ACE 06)*, pp. 157–163 (2006).
- [2] 上野 真, 照井佑季, 今村瑠一郎, 久野 靖, 江木啓訓: 短冊型プログラミング問題における解答プロセスに基づく指導支援, マルチメディア, 分散協調とモバイルシンポジウム 2021 論文集, pp. 1451–1458 (2021).
- [3] Nakayama, Y., Kuno, Y. and Kakuda, H.: Split-Paper Testing: A Novel Approach to Evaluate Programming Performance, *Journal of Information Processing*, Vol. 28, pp. 733–743 (2020).
- [4] 山口 琢, 中村陽太, 大場みち子: プログラム・コードの並べ替えパズルにおける正解との距離の変化, 情報教育シンポジウム論文集, pp. 47–53 (2020).
- [5] Du, Y., Luxton-Reilly, A. and Denny, P.: A Review of Research on Parsons Problems, *Australasian Conference on Computing Education ACE'20*, pp. 195–202 (2020).
- [6] Cheng, N. and Harrington, B.: The Code Mangler: Evaluating Coding Ability Without Writing Any Code, *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pp. 123–128 (2017).
- [7] Zhi, R., Chi, M., Barnes, T. and Price, T. W.: Evaluating the effectiveness of parsons problems for block-based programming, *2019 ACM Conference on International Computing Education Research*, pp. 51–59 (2019).
- [8] 藤井沙苗, 松澤芳昭: パズル型問題を利用したプログラミング初学者の理解度と思考過程の分析, 情報処理学会第 83 回全国大会 (2021).
- [9] 山口 琢, 松澤芳昭, 新美礼彦, 大場みち子: 取捨選択操作の時間的な共起分析によるプログラミング・プロセスでの迷いの検出, 情報処理学会研究報告 (CE-160-8) (2021).
- [10] 佐藤隆博: S-P 表の入門, 明治図書出版 (1985).