

# アルゴリズム構築を指向した プログラミングパズル問題の開発と実践

藤田然<sup>†</sup> 金森大輝<sup>†</sup> 松澤芳昭<sup>†</sup> 鈴木彩未<sup>†</sup>

青山学院大学社会情報学部

## 1 はじめに

我々は、パズル型プログラミング問題を使った学習者の思考過程の分析の研究を行ってきた。本研究では、各講義の教材からどこを問うべきかを議論しなおし、既存問題 [1] の改良と新たなパズル型プログラミング問題の開発を行った。最終目標は、正答率や、回答パターンなどの分析を通じて、問題の良し悪しを評価し、難しすぎず、受講者の理解を深めアルゴリズム構築能力を育成する良質なパズル問題を受講生に提供することである。

## 2 方法

### 2.1 ジグソー・コード

本研究で扱うパズル問題は、ジグソー・コードを用いて開発されている。ジグソー・コードは、山口らが開発を進めているパズル型問題の実験システムである [2]。

### 2.2 問題作成・実施方法

実験の対象は、青山学院大学社会情報学部社会情報学科一年生で「コンピューティング実習」の受講者約 200 人である。小テストとして毎回の単元の範囲を理解しているかを確認するために実施した。

問題作成をする際は、コンピューティング実習の教材を元に、どこを問うべきか、正答率をどのくらいに設定するかを議論を行った。作成した問題の正答率予想や想定される間違い選択肢の割合もあらかじめ予想した。

パズル型問題 24 問を表 1 に示す。表 1 中、「番号」は問題番号、「単元」はその問題の主題、「問題名」は各問の問題の名前、「識別」、「困難」は項目反応理論

表 1: パズル型問題一覧

番号	単元	問題名	識別	困難	知	ア
P3.1	条件分岐	くじびき	0.51	0.47	2	1
P3.2	条件分岐	今日傘いる	0.37	0.29	1	1
P3.3	条件分岐	Youtuber	0.30	0.87	3	1
P4.1	繰り返し	三菱	0.31	0.54	0	2
P4.2	繰り返し	雪の結晶	0.44	0.27	0	2
P4.3	条件分岐	割引料金	0.33	0.57	2	2
P4.4	繰り返し	サクラ	0.45	0.48	0	2
P5.1	繰り返し	3 が出た数	0.41	0.13	1	2
P5.2	入れ子構造	雪の結晶 2	0.52	0.34	0	2
P5.3	入れ子構造	ツリー	0.49	0.21	0	2
P5.4	入れ子構造	九九	0.27	0.13	0	3
P6.1	関数 (引数)	さくらんぼ	0.46	0.30	0	1
P6.2	関数 (引数)	星を 7 個	0.53	0.70	2	0
P6.3	関数 (引数)	晴れマーク	0.62	0.46	1	0
P6.4	関数 (引数)	蜂の巣	0.42	0.64	1	0
P7.1	関数 (戻り値)	還元率	0.42	0.47	2	0
P7.2	関数 (戻り値)	ラッキー 7	0.49	0.38	1	0
P8.2	関数 (引数)	風車	0.39	0.68	0	2
P11.1	条件分岐	ガチャ	0.38	0.61	1	1
P11.2	条件分岐	おつかい	0.39	0.32	1	1
P11.3	繰り返し	偶数リスト	0.43	0.32	1	1
P12.1	条件分岐	挨拶	0.21	0.34	0	2
P12.2	繰り返し	表作成	0.29	0.29	1	1
P12.3	繰り返し	階段	0.36	0.12	1	1

(IRT) を用いて得た識別度と困難度、「知」、「ア」は各問のプログラミング知識、アルゴリズムそれぞれの想定誤答パターンの数を示している。

### 2.3 分析方法

得られたデータを元に、正答率と想定される間違い選択肢の割合を調べ、予想した正答率や想定される間違い選択肢の割合との差を分析した。また、項目反応理論 (IRT) を用いて問題の分析を行った。

## 3 結果

### 3.1 IRT による分析結果

パズル問題の解答結果より作成した項目特徴曲線を図 1 に示す。全体としては、グラフの形はおおよそは似通っているため、差はあれど問題の識別度はほぼ同じくらいである。しかし、P3.3、P4.1、P4.3、P5.4、P12.1、P12.2 に関しては、他のグラフとは異なっているため、識別度が低い。これは、表 1 からとも言える。

## Practice and Report of Programming Puzzle Questions for Algorithm Construction

Zen FUJITA<sup>†</sup>, Daiki KANAMORI<sup>†</sup>, Yoshiaki MATSUZAWA<sup>†</sup>, Ayami SUZUKI<sup>†</sup>School Of Social Informatics, Aoyamagakuin University  
252-5258, sagamihara, Japan

matuzawa@si.aoyama.ac.jp

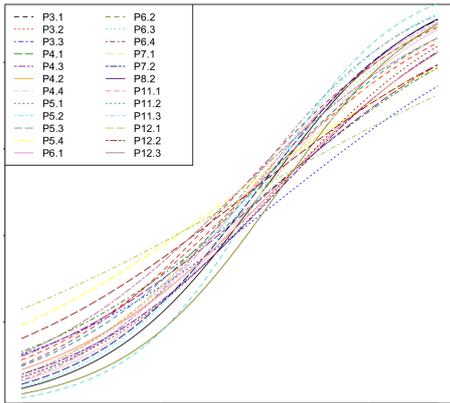


図 1: IRT による項目特性曲線

```

s1 T for (var i = 1; i <= 9; i++) {
s2 T   for (var j = 1; j <= 9; j++) {
s3 T     print(j * i + ',');
s4 T   }
   println('');
}
    
```

図 2: P5.4 「九九」

識別度の低い問題の例として、P5.4「九九」の問題を図2に示す。また、P5.4「九九」の意図と解答パターンを表2に示す。この問題は、4つのピースを並び替える問題であり、比較的正答率の高い問題になるだろうと予想したが、結果としては13%であり、識別度が低く、困難度が高い問題となった。

このような結果になった原因として考えられることは、二つある。一つは、問題の意図として、アルゴリズムの問いを多くし過ぎてしまったことである。この問題はアルゴリズムによる間違いパターンを3つも置いてしまったことである。二つ目は、「print」と「println」の細かい違いを出したためである。「print」と「println」は授業の第二回目に出した内容であり、その後はそれらについて触れていなかったため、できる人も間違えてしまったのではないかと考察した。

### 3.2 学生の理解度の分析

パズル問題の解答結果より作成した単元別の正答率を図3に示す。グラフを見ると「入れ子構造」次いで「繰り返し」のようなアルゴリズムの問題の正答率が低いことが言える。特に入れ子構造は、P5.4「九九」のように入れ子構造に加えて他の問いを追加することによって、問題の困難度が高くなることがわかった。

繰り返しは、通常の問題の困難度は普通ぐらいであるが、P12.2の繰り返しによって表を作る問題や、P12.3の canvas の clear を使うようなアルゴリズムとプログ

表 2: P5.4 「九九」 解答分類

正答	13%
print と println の違いを理解しているか (アルゴリズム)	27%
入れ子の構造を正しく理解できているか (アルゴリズム)	21%
一重ループになっている (アルゴリズム)	23%
その他	12%

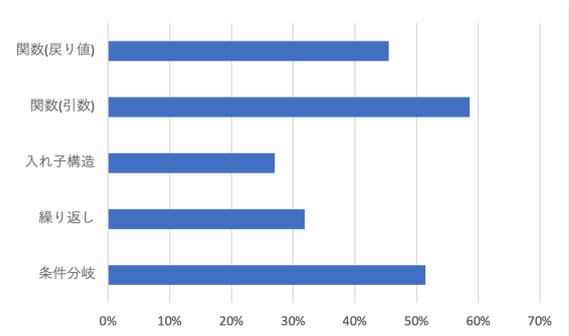


図 3: 単元別の正答率

ラミング知識を混ぜた問題が困難度が高くなっている。実際、その知識を問う問題を実施したところ 35%程度の人が予想していた不正解をした。以上の結果から、入れ子構造や繰り返しは、アルゴリズムやプログラミング知識を混ぜることによって困難度が高くなりすぎると考えられる。

繰り返しで困難度が高くなっている問題は P4.2 である。P4.2 は、繰り返し内の変数の最初の値を問う問題である。正解は、「i=0」であるが、実際の解答の多くは「i=1」であった。この原因としては、繰り返しの文がどのように繰り返されているかについて理解ができなかったことが考えられる。しかし、教材を見ると別の要因があったとも考えられる。我々が今回作成した問題の繰り返しは基本的に「i=0」で作成した。だが、教材では「i=1」で統一されておりその違いにより生徒が普段書いている「i=1」を選んでしまったと考えられる。そのため、学生がどちらで間違えているのか正確に問題で測ることができなかった。これらを判断するためには、繰り返し内の変数を使うような問題を作成する必要があると考察できる。

### 参考文献

[1] 西村萌, 米澤彩乃, 上島綺夏, 黒木奏子, 松澤芳昭: "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", 情報処理学会研究報告 CE-164 No.7, 2022.

[2] 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よる正解との距離の変化", 情報教育シンポジウム 2020 論文集, pp.47-53, 2020.