

情報リテラシーおよびプログラミングに関する 理解度テストの実施と分析

Practice and Analysis of Comprehension Tests for Information Literacy and Programming

野崎 崇弘†

村川 猛彦†

Takahiro Nozaki

Takehiko Murakawa

1. はじめに

大学（学部）の授業は、セミナーのような科目と卒業研究を除けば、講義と演習に大別される。講義は1名の教員が話す（いわゆるオムニバス科目として、1科目を複数の教員が担当することがあっても、各回の授業で話すのは1人である）のを多数の学生が同時に聴いて学ぶのに対し、演習（実験・実習を含む）は教員の与える課題に各学生が取り組んで成果を得るものとなる。

講義科目の成績評価には、試験とレポート、およびその組み合わせが用いられ、小問の解答や、授業を踏まえて自分なりに課題を設定し答案にまとめることができるかを通じて評価される。演習科目において、成績評価の対象は成果物（プログラミング課題であればソースコードなど）が中心となる。その場合、学生はしばしば、個別の課題の答えを求めることに時間が取られ、科目を通じて学ぶべき事項の見直しと振り返りが十分に行えないこととなる。

本研究では、学習事項の定着の支援を目的として、理解度テストを改良または新規に開発した。テストは3種類で構成され、情報リテラシーの理解度テスト（2節）、Javaの正誤判定テスト（3節）、およびJavaのforループ読解テスト（4節）である。情報リテラシーの理解度テストは和歌山大学システム工学部1年生向けの科目を対象とし、2017年度および2018年度に実施したテストの改良版であり、ある授業回で解答してもらい、文献[1]と同様に項目反応理論に基づき分析を行った。Javaに関する2種類のテストは新たに開発し、同3年生向けの演習科目で、Javaプログラミングの基礎となる知識および技術の理解・復習を念頭に置き、同一問題や類似問題を入れながら数回実施し集計を行った。

2. 情報リテラシーの理解度テスト

2.1 対象科目およびテストの概要

和歌山大学システム工学部の1年生向け情報処理教育には情報処理Ⅰ・情報処理Ⅱの2科目があり、それぞれ前期と後期に開講の必修科目である。このうち情報処理Ⅰは情報処理リテラシーを学習する科目であり、学内計算機環境やインターネットリソースなどを適切に使用するための知識や、文書作成・データ管理・プレゼンテーションを行うためのアプリケーションの使い方、ハードウェア・ソフトウェア・クラウドサービスを含む情報システムの仕組みについて学ぶ。

各授業は、教員による解説と計算機を用いた演習課題で構成されており、復習を通じた、学習事項の定着の機会がほとんどなかった。そこで、時間外学習を通じて学んだ内容の見直しと振り返りができるような、問題セットの開発および学習支援環境の提供が望まれる。

情報リテラシーに関する最初の理解度テストは、2015年度の情報処理Ⅱの初回の授業で実施し、5択問題10問が含まれている。2016年度にも同様の形式で5択問題を20問に増やした。いずれも紙上で解答してもらい、答え合わせののち授業中に回収した。また2016年度後期には、情報処理Ⅰ関連で133問、情報処理Ⅱ関連で100問を作成するとともに、Webアプリケーションを構築し、情報処理Ⅱの受講者に使用してもらった[2]。これにより1人当たりの平均解答数や思考時間などの数量情報を得たが、1問当たりの解答者数は十分ではなく、各出題の良し悪しを検討するのは困難であった。2017年度以降は情報処理Ⅰの（全15回のうち）第14回の授業で、Moodleの小テスト機能を用いて5択問題を解答してもらい、項目反応理論に基づく分析を試みた[1][3]。また共通項目計画に基づく等化を実施し、2017年度の理解度テスト解答者を基準集団として、2018年度実施分の等化係数を求めて各解答者の等化後の能力値を算出し、2年度分の能力値を同一散布図上に重ね合わせた。2017年度、2018年度、2019年度において分析対象の解答者数は順に、101名、129名、124名であった。

オフィススイートやブラウザなど、ソフトウェアの活用を含むパフォーマンス評価は、本研究の対象外となる。検索エンジンなどを効果的に使いながら、情報を批判的に精査して正確なウェブ上の情報を収集するための能力を、ウェブアクセスリテラシーと定義し、その能力を測定するための尺度および質問紙の提案ならびに分析が[4]に見られる。

分析に使用した項目反応理論[5]について手短かに説明する。正答率や素点に基づくテスト（古典的テスト理論と呼ばれる）では、正解率が高かった問題について、問題が易しかったからか、それとも解答者の能力が高かったからなのかを判断できない。それに対し、適切な計画を立ててテストを行い、項目反応理論を用いて分析を行うことで、解答者の能力に関する情報と、各出題（項目）の難易度などに関する情報とを、切り離して推定することが可能となる。分析にあたり本研究では、行を解答者、列を出題として、正答には1、誤答には0を格納した表（項目反応パターン）を作成し、統計解析ソフトウェアRへの入力とした。なお、パラメタ推定を行うためにirtosパッケージを、また等化に関してはplinkパッケージを使用した。各解答者の能力値は1個の実数値で表される。また項目の推定にあたり、識別力と困難度という2つの実数値に基づく2パラメタ・ロジスティックモデル（2PLM）を採用した。識別力と困難

†和歌山大学, Wakayama University

度の推定値が得られれば、2次元平面上に項目特性曲線と呼ばれる曲線を描き、他の項目との比較ができる。識別力の値は高い（項目特性曲線は中心に大きな傾きを持つ右上がりのS字を描く）ほうがよい。それに対し困難度は高ければよいとは限らず、基本的知識を測る理解度テストにおいては、低い困難度は「きちんと聞いていたか」を把握するための設問として有用であると言える。

項目反応理論を用いることで得られる他の利点として、異なるテストの結果を同一の尺度に乗せ、比較可能にできる点が挙げられる。その手続きは等化と呼ばれる。異なるテストを比較できるよう、本研究では、テスト間に共通する問題が含まれている「共通項目計画」を採用した。線形等化関数を用いた場合、一次変換の傾きと切片という2つの実数値を得ることができ、合わせて等化係数と呼ばれる。

2019年度に実施した理解度テストでは、共通項目計画に基づく等化を行うことを念頭に置き、2018年度に実施した理解度テストを基に問題を追加・削除することとした。追加にあたり[6]で自動生成した問題から選び、一部を編集した。2018年度の実施において、識別力が低く分析の対象とすることができない問題が含まれており、今年度においても同様の識別度の低い問題が含まれることが予想されるため、それまでの20問から25問へと問題数を増やした。全てを解答してもらう方針は、変えていない。25問のうち、2018年度と出題内容の一致する問題は10問とした。

2.2 実施結果および分析

2019年7月の情報処理Iの第14回の授業で、2教室の受講者に解答を依頼した。25問全てに解答し、適切に記録が保存された124名の解答をもとに、項目反応パターンを作成し、[1]と同様に分析を行った。その解答状況を表1に示す。また問題除去前の識別力・困難度をもとにした25問の項目反応曲線を図1に示す。横軸は能力値、縦軸は正答確率（その能力値の解答者集団のうちどれだけの割合が正答するか）を表す。

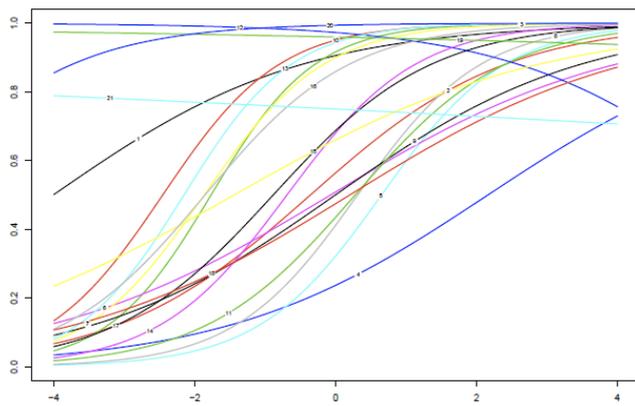


図1 情報リテラシー理解度テストの項目反応曲線

問題番号4, 8, 12, 15は正答率と識別力が高く、困難度は低いため、「きちんと聞いていたか」を把握するための設問として有用であると考えられる。最も識別力の高い問題番号4は、2019年度に新たに解答してもらったものであり、「単位時間あたりのデータ転送量は転送速度と呼ばれ、【 】といった単位で表される。」（正解はbps、誤選択肢はbit, raw, bmp, gif）という問題であった。

表1 情報リテラシー理解度テストの問題ごとの状況

問題番号	正答率	問題除去前		問題除去後	
		識別力	困難度	識別力	困難度
1	0.508	0.196	-0.167		
2	0.895	0.564	-4.011		
3	0.556	0.698	-0.369	0.714	-0.361
4	0.863	1.387	-1.743	1.322	-1.796
5	0.250	0.507	2.286	0.499	2.320
6	0.355	1.174	0.638	1.206	0.630
7	0.508	0.496	-0.073	0.495	-0.073
8	0.855	1.184	-1.859	1.230	-1.815
9	0.427	1.115	0.316	1.182	0.306
10	0.363	0.227	2.507		
11	0.500	0.586	-0.006	0.578	-0.005
12	0.919	1.226	-2.427	1.215	-2.446
13	0.444	0.939	0.276	0.977	0.270
14	0.968	-0.615	5.826		
15	0.903	1.313	-2.148	1.186	-2.300
16	0.645	1.147	-0.673	1.101	-0.688
17	0.653	0.421	-1.570	0.432	-1.532
18	0.484	-0.227	-0.289		
19	0.823	1.018	-1.798	0.962	-1.873
20	0.661	0.878	-0.896	0.824	-0.938
21	0.718	-0.233	4.054		
22	0.476	0.476	0.210	0.455	0.220
23	0.960	-0.137	23.219		
24	0.992	0.640	-7.820		
25	0.750	-0.078	14.105		

文献[1]と同様に、識別力が0.3未満、または困難度の絶対値が4を超えるものを除去することで、等化における共通項目は、2019年度の問題番号6・7・9・16・17と2018年度の問題番号4・5・9・10・18の5問となった。基準集団を2019年度の理解度テスト解答者とし、等化法にはMean & Sigmaを指定して、Rのplinkパッケージを用いて計算した結果、等化係数として、傾き $A=0.779628$ および切片 $K=0.148330$ を得た。

この値を用いて2018年度実施分について等化後の項目パラメタ（識別力および困難度）ならびに各解答者の能力パ

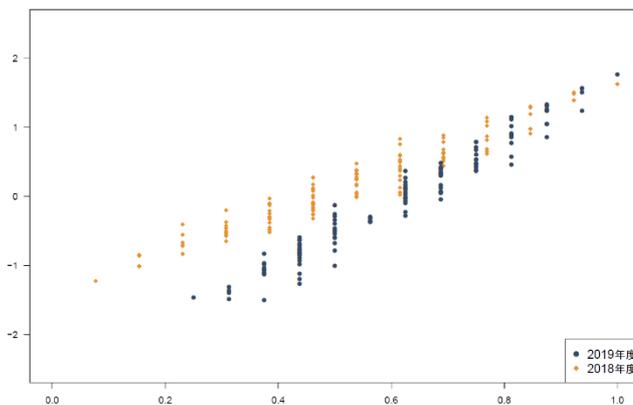


図2 正答率と等化済み能力値の散布図

ラメタを算出した。2018 年度実施および 2019 年度実施について、正答率および能力値の散布図を図 2 に示す。この図において横軸は正答率（2018 年度は正答数÷13，2019 年度は正答数÷16），縦軸は等化後の能力値を表す。2018 年度的全問正解者は 2 名，2019 年度は 1 名であった。

2018 年度および 2019 年度実施分の比較を行う。2019 年度の方が全体的に正答率は高く，また正答率が 0.8 以上かつ除去されていない問題が存在している（2018 年度には得られなかった）。等化を行った結果に関しては，図 2 の正答率中下位（0.750 以下，なおいずれも除去後の問題の正答率であるため，実際（2018 年度は 20 問，2019 年度は 25 問）の正答率とは異なる）を見る限り，2018 年度の解答者のほうが能力値は総じて上位になっている。

2018 年度は 20 問中 7 問，2019 年度は 25 問中 9 問を分析対象から除去する必要があった。問題数を増やすことで分析可能な問題を増やすことはできたが，除去対象となる問題が多く発生しているという状況は変わっていない。

3. Java の正誤判定テスト

3.1 対象科目の概要

本節および次節のテストを実施した授業科目，およびテスト実施の経緯について述べる。科目名は Web アプリケーション構築演習であり，筆者のうち一人（村川）が 2017 年度より担当している授業である。和歌山大学システム工学部ネットワーク情報学メジャーの 3 年前期選択科目として開講し，同メジャーのほか知能情報学メジャーを第 1 メジャーとする学生の履修が大部分を占める。受講者数は各年度 30~40 名程度である。

2019 年度は，各受講者が持参する BYOD (Bring Your Own Device) の PC に，JDK (Java Development Kit) や Apache Tomcat (サーバーソフトなどの実行が可能な HTTP サーバ) を含む Eclipse (統合開発環境) をインストールし演習課題に取り組んでもらった。Java を使用してサーバサイドの Web アプリケーションを作成することを原則とし，フロントエンドにおける JavaScript の使用については一部の課題に限って認めた。

ネットワーク情報学メジャーのカリキュラムにおいて，2 年後期にオブジェクト指向プログラミング演習という選択科目があり，多くの学生はここで初めて Java の文法ならびに Java を用いたプログラムの書き方を学習する。一方，Web アプリケーション構築演習の履修を前提とする科目は他になく，研究室で使用するのでなければ，学生にとって Java の学習はこの科目が最後となる。

Web アプリケーション構築演習においても，情報処理科目と同様にはじめの 2 年間は，教員より課題を提示し，授業中に（ときには授業時間外にも）取り組んで答案を提出してもらい，その内容で成績評価を行ってきた。

筆者らは，Web アプリケーションのみならず Java のクラスを自分で設計・実装・テストする能力や，API の活用，ならびに Java の言語仕様や JVM (Java 仮想マシン) の変化を含む近年の動向について，自習し，または必要に応じて調査し活用できる能力も，学部生が身につけることを期待する。しかしながら，カリキュラムや授業時間数といった制約を考慮すると，座学（演習科目中の教員による解説を含む）を通じて身につけさせるのは困難であるとも考えている。

3.2 テストの概要

上に述べた課題のもとで，効果的な知識獲得を支援するため，〇×クイズによる情報の提供を試みた。主な目的を，継続的な解答の機会を通じて Java の基礎知識を定着することとし，出題対象は授業の課題内容に限定しなかった。具体的には，コマンドラインで使用するコマンドや，Java 7 以降で利用可能となった，switch 文の値に文字列を指定できることなどを取り入れた。

問題開発にあたり，正誤判定問題とし，1 つの対象で正誤の問題を 1 問ずつ作成した。ペアの例を示す。

(正) 配列 a の要素数は a.length，文字列 s の長さは s.length() で，それぞれ求められる。
(誤) 配列 a の要素数は a.length()，文字列 s の長さは s.length で，それぞれ求められる。

最初の出題までに 87 ペア，計 174 問を作成し，その後は，増減や問題文の修正をしなかった。

授業での出題は，第 1 回から第 8 回まで実施し，1 回の授業につき 20 問とした。まず 87 ペアからランダムに 20 ペアを抽出し，各ペアにおいて，正:誤=35:65 の比で，いずれかの問題を取り出した。誤の割合を高くしたのは，批判的に文章を読むことを期待したためである。問題の抽出，および Moodle のインポートに適したフォーマットの出力は，自作の Ruby スクリプトが処理しており，同一問題や，同一ペアの異なる問題が，別々の実施回で出題されるようになっている。実施回の間には，依存や排他（ある回で出題したら，次回は出題しないなど）はない。また正誤の比は処理スクリプトにおける乱数生成に基づいており，20 問のうち正になるものを 7 問，誤になるものを 13 問としたわけではない。

解答は，授業の課題や解答を管理している学内運用の Moodle の小テスト機能を用いて実施し，20 問すべてに解答した後に，それぞれ正解か不正解かを表示するようにした。また解答ごとに Moodle サーバに結果が保存され，授業後に Excel ファイルとして回収した。この解答は成績に入れないこと，および解き直しをしないことを授業中に指示した。結果の分析にあたっては，解答途中のものや同一ユーザの複数回解答を手で除去してから，解答者情報を通し番号に置き換えて匿名化し，分析処理を実施した。

3.3 実施結果および分析

実施回ごとの解答者数および正答率を表 2 に示す。

実施回	解答者数	問題の正答率		
		平均	最高	最低
1	24	0.54	0.91	0.18
2	28	0.58	1.00	0.24
3	28	0.53	0.97	0.17
4	24	0.59	0.96	0.14
5	28	0.62	0.96	0.25
6	25	0.58	0.96	0.08
7	27	0.55	0.93	0.22
8	27	0.58	0.89	0.07

解答者の数量情報を述べる。1回以上解答したのが30名で、このうち19名が8回すべてに解答している。すべてに解答しなくても10問以上正解だったのは5名で、その最小正解数は10問または11問であった。すべての実施回および解答者を通じて、20問全問正解はなく、最大正解数は19問、最小正解数は6問であった。

出題された問題に関して、87ペアのうち5ペアが4回出題され、逆に8ペアは1回も出題されなかった。第8回に正解率が0.07だった問題も、4回出題されたうちの一つであり、4回とも誤が正解であった。具体的には以下の文である。

Java のコードエディタやビルドツール、ソフトウェア開発を支援するソフトウェアを一つにまとめたパッケージはJDKと呼ばれる。

JDKには、Java コンパイラ (javac) や Java アプリケーションのローダ (java) が含まれているが、Ant や Maven に代表されるビルドツールも、コードエディタも含まれていない。この出題のペアで正の文は「Java のコンパイラやローダなど、ソフトウェア開発を行うのに必要なソフトウェアを一つにまとめたパッケージはJDKと呼ばれる。」であった。

この出題に関して、出題した回と問題番号、正答率を見ていくと、第2回は第3問で0.24、第4回は第2問で0.22、第5回は第3問で0.25、そして第8回は第1問で0.07であった。前述の通り全てを解答してから全問の正誤結果が表示されること、この出題はいつでも早い番号で出題されていたこと、なぜ誤答となるかの解説を付けていなかったことから、「何回出題しても学習効果がなかった」出題と考えられる。各実施回の正答率の平均が、ほとんど変化していないのも、学習効果がなかったことを示唆する。

今後の改善にあたっては、抽出した順に出題するのではなく出題順を変えること、誤答に対し解説を設けること、学生に学習の動機付け (回数を重ねれば正答数が増える) となるような出題や指示をすることなどが挙げられる。

4. Java の for ループ読解テスト

4.1 テストの概要

前節の正誤判定問題は、Java に関する個別の知識を問うものである。宣言や式評価といった、プログラムコードの理解に関する出題もいくつか作成したが、一つのプログラムまたは複数行のコードを与え、どのような処理をするのかを出題するのは、問題文の見せ方を含む出題管理において、適切とは言えない。

そこで正誤判定と別に、数行で記述された for ループについて、変数の値がどのように変化し、反復において順に何を出力するかを解答してもらい、処理を正しく理解しているかどうかを確認できる問題を作成した。出題におけるプログラムサイズ (行数) は、変数の数と for 文の数の組み合わせで異なる。プログラムコードから容易に処理内容を理解できるかという観点で、タイプ I (1 変数・1 つの for 文)、タイプ II (2 変数・1 つの for 文)、タイプ III (2 変数・2 つの for 文、2 重ループ) の 3 種類を用意した。それぞれの例を以下に示す。

タイプ I (1 変数・1 つの for 文)

```
for (int i = 1; i < 10; i *= 2) {  
    System.out.println(i);  
}
```

タイプ II (2 変数・1 つの for 文)

```
for (int i = 1, j = 1; j < 30; i++, j *= i) {  
    System.out.println(i + ", " + j);  
}
```

タイプ III (2 変数・2 つの for 文、2 重ループ)

```
for (int i = 1; i <= 3; i++) {  
    for (int j = 1; j < i; j++) {  
        System.out.println(i + ", " + j);  
    }  
}
```

1 題 (コラッツ予想に関連する for ループ) を除き、ループ内の処理は、その時点の変数の値を、System.out.println メソッドを呼び出して出力するのみとした。呼び出しごとに改行することになるが、解答時に改行は空白文字に置き換えて 1 行で解答するよう指示し、答案回収後に手で正誤判定を行い、改行を用いた答案も正答とした。前述の 3 つの例題に対し、正解は順に、「1 2 4 8」「1,1 2,2 3,6 4,24」「2,1 3,1 3,2」である。

10 問作成し、Moodle の小テスト機能を用いて、Web アプリケーション構築演習の第 1 回から第 10 回まで 1 問ずつ解答してもらった (類似問題はあるが、同一問題はない)。解答後に、正誤を表示する (正解文字列との一致判定のため、改行を入れた解答などは誤答として表示される)。どのプログラムコードも、for 文の初期化で、ループを有効範囲とする変数宣言を行っている。そのため、Java 9 以降の JDK に含まれている jshell コマンドを起動し、プログラムコードをそこに貼り付ければ、正解となる出力を得ることができる。jshell は解答時に使用しないこと、前節と同様にこの解答は成績に入れないこと、および解き直しをしないことを指示した。

今回解答してもらった学生にとって、for ループの学習は Java プログラミングの科目よりも前に行っている。具体的には、まず情報処理 II の授業で、C 言語による for ループを学習しており、九九の表を生成するプログラムにおいて、独立型 (内側のループ用変数がかかる値の範囲が、外側のループ用変数に依存しない) の 2 重ループを含むプログラムを作成している。2 年の以降はメジャーごとに異なる科目となるが、ネットワーク情報学メジャーまたは知能情報学メジャーを第 1 メジャーとする学生は、2 年前期にアルゴリズム演習 I という必修科目を通じて、for ループを含む C 言語の多様なプログラム例を経験している。

4.2 実施結果および分析

問題ごとの正答率を表 3 に示す。

表3 Javaのforループ読解テストの結果

実施回	タイプ	解答者数	正答率
1	I	34	0.68
2	III	29	0.76
3	III	28	0.68
4	I	24	0.96
5	II	25	0.60
6	I	24	1.00
7	III	25	0.64
8	I	23	0.52
9	II	26	0.73
10	II	26	0.65

タイプIで正答率の最も高い問題は第6回で全員が正解した。これはi--を用いて1ずつ減らすというforループであり簡単すぎたと考えられる。正答率の最も低い問題は第8回で、半数近くが間違えていた。これはコラッツ予想に関するforループで、ループ用変数の値の変化は、for文ではなく内部処理の中で(奇数ならば3倍して1を加え、偶数ならば2で割るという処理により)行っていたこと、そしてi++でもi--でもなかったことから、戸惑いがあったと思われる。

タイプIIについてはいずれも正答率はやや低くなった。解答者にとって馴染みが薄いためと考えられる。演習課題の解答として提出されたプログラムで、この処理を使用したものはなかった。

タイプIIIの中では第7回が最も低かった。前述の例で示したプログラムであり、「int i = 1」の宣言にもかかわらず、i=1のときの値を出力しないことが、間違いやすい原因と考えられる。誤答の9人のうち7人が「1,1」から始まる解答をしていた。

今回の出題および分析を通じて、forループを通じたプログラミング読解能力把握の足がかりを得ることができた。ただし問題の数や質(類似問題)については、充実および見直しを図る必要がある。また使用言語について、C言語で同等のプログラムコードを作ることとも可能である。System.out.printlnの呼び出しをprintf関数に置き換えるとともに、変数宣言をforループより前に行うよう、書き換えればよい。

5. おわりに

本稿では、和歌山大学システム工学部1年生を対象とした情報リテラシーの理解度テスト、および3年生を対象としたJavaに関する2種類のテストの実施と分析結果について述べた。理解度テストでは、「きちんと聞いていたか」を把握するための設問として有用と思われるものを得ることができた。Javaの正誤判定テストは、全体の平均正答率、および何回か実施した同一問題の正答率の推移から、「何回出題しても学習効果がなかった」ことが判明し、学びにおけるフィードバックの必要性を痛感することとなった。Javaのforループ読解テストは、単ループおよび2重ループにおいて、さらなる出題の拡充が求められる。

今後の課題としては、各テストの節で書いた事項の他に、情報処理Iを履修する1年前期と、Webアプリケーション構築演習を履修する3年前期の間の、学生の理解状況の調査および比較が挙げられる。情報リテラシーの理解度につ

いては追跡調査を実施すること、またforループの読解に関しては、学び始めの1年後期、経験を得ていく2年前期・2年後期の学生に、(C言語で記述した)同一問題を解いてもらい、今回の結果と比較することが望まれる。

謝辞

理解度テストの解答に協力してくださった受講生の皆様に感謝申し上げます。情報リテラシーの理解度テストの準備にあたっては、岡野元春氏より、またJavaの正誤判定テストの集計に関しては、東涼介氏ならびに村田亘平氏より、それぞれ協力を得ました。ここに御礼を申し上げます。

本研究はJSPS科研費19K12267の助成を受けたものです。

参考文献

- [1] 松本陽平, 村川猛彦: 情報リテラシーを対象とした理解度テストの生成と分析, 2018年度情報処理学会関西支部支部大会, E-04 (2018).
- [2] 松本陽平, 藤原敬介, 村川猛彦: 情報処理教育を対象としたeラーニングシステムの構築, 情報知識学会誌, Vol.27, No.2, pp.155-160 (2017).
- [3] 村川猛彦, 登田純太, 松本陽平: 項目反応理論に基づく大学初年度向け情報リテラシー科目の理解度テスト分析, 2018年電子情報通信学会総合大会, 情報・システム講演論文集1, p.155, D-15-23 (2018).
- [4] 山本祐輔, 山本岳洋, 大島裕明, 川上浩司: ウェブアクセスリテラシー尺度の開発, 情報処理学会論文誌データベース(TOD), Vol.12, No.1, pp.24-37 (2019).
- [5] 加藤健太郎, 山田剛史, 川端一光: Rによる項目反応理論, オーム社 (2014).
- [6] 松本陽平: 情報リテラシーを対象とした理解度テストの開発, 和歌山大学大学院システム工学研究科修士論文 (2019).