Vol.2021-CE-161 No.1
2021/10/23

Googleフォームを利用したプログラミング問題の生成システムの開発と評価

大門 巧[1]　大西建輔[2]

概要：高等学校において教科としてプログラミングを教える場合，評価をおこなう必要がある．教員の働き方改革が推進されている現在，問題作成や採点の効率化を図ることは喫緊の課題となっている．我々は，プログラミング問題の自動生成及び自動採点システムである Waquema を開発した．Waquema には，ウェブページを出力するウェブ版と Google フォームを出力するクラウド版がある．著者の所属する大学の講義において，クラウド版で生成した問題を演習問題や試験問題に使用した．学生に対してアンケートを実施し，システムの評価をおこなった．また，使用した教員へアンケートをおこない，演習課題や試験にかかる時間の評価をおこなった．その結果，生成された問題の難易度は学生にとって適切であり，本システムを用いて数をこなす必要がある演習問題を作成する場合に，時間の短縮につながる可能性があることが分かった．

キーワード：プログラミング教育，問題生成，自動採点

Evaluation of a System for Generating Programming Problems with Google Forms

Takumi Daimon[1]　Kensuke Onishi[2]

Abstract: Development and Evaluation of a Programming Problem Generation System Using Google Forms When programming is taught as a subject in high schools, it is necessary to evaluate it as a subject. In order to improve the efficiency of question generation and grading, it has become an urgent issue. We have developed Waquema, a system for automatic generation and grading of programming problems, which has two versions: a web version that outputs web pages and a cloud version that outputs Google Forms. We used the questions generated by the cloud version of Waquema for exercises and examinations in lectures at the university to which the author belongs. We conducted a questionnaire survey on students to evaluate the system. A questionnaire was also sent to the teachers who used the system to evaluate the time required for the exercises and examinations. The results show that the difficulty level of the generated problems is appropriate for the students, and that the system may save time when they prepare exercises.

Keywords: Programming Education, problem generation, automatic grading

1. はじめに

学習指導要領の改訂に伴い，2022 年度より高等学校では科目として「情報Ⅰ」と「情報Ⅱ」が新設される．必履修となる情報Ⅰのコンピュータとプログラミングの単元では，プログラミングを学習する．教科としてプログラミングを教える場合，評価をおこなう必要がある．そのため，プログラミング言語やアルゴリズムの理解度の測定や問題演習が必要となる．この際，教員が作成したテストやオンラインジャッジシステムを用いることが考えられる．教員がテスト問題を作成する場合，筆記式のテストや学習管理シス

[1] 東海大学大学院理学研究科数理科学専攻
Tokai University Graduate School of Science Course of Mathematical Sciences, Kanagawa, 259-1292, Japan
[2] 東海大学
Tokai University, Kanagawa, 259-1292, Japan

ⓒ 2021 Information Processing Society of Japan

テムを利用し，コンピュータ上でおこなうテストを作成することになる．

筆記式のテストは，コンピュータを使用せずにおこなうことができるという利点があるが，教員が出題する問題を熟考することや採点に時間を割かれてしまう．教員の働き方改革が推進されている現在，問題作成や採点の効率化を図ることは喫緊の課題となっている．

オンラインジャッジシステムでは作成したプログラムを提出すると，簡単に問題の正誤判定をおこなうことができるという利点がある．一方で，オンラインジャッジシステムにおいて出題されている問題は，解答例と出力の一致するプログラムを実装する問題がほとんどである．理解度を高めようとする学習者にとって，学習中のテーマに沿った様々な問題がランダムに出題されるような状態であることが望ましい．

プログラミング問題の生成・採点を自動でおこなうシステムは，Java 言語を中心としたものが散見され，大学生や大学教員が自作したシステム [3] が大学を中心に利用されているが，学習中のテーマに沿ったランダムな問題が生成されるシステムは見られない．本研究では，高等学校の情報 I，II で教育用に用いられる教育用のプログラミング言語として，Python と JavaScript の両方に対応したプログラミング問題の自動生成及び自動採点システムである Waquema[1] を開発した．Waquema には，ウェブページを出力するウェブ版と Google フォームを出力するクラウド版がある．本論文では，クラウド版のシステムの開発と評価について述べる．

本論文の構成は次の通りである．2 節では，この分野の関連研究を紹介する．3 節では，本システムの概要について述べる．4 節では本システムで生成可能な問題について，5 節では実施したアンケートの詳細とそのデータについて説明する．6 節では，アンケート結果についての考察をおこなう．7 節では本論文のまとめを述べる．

2. 関連研究

2.1 プログラミング問題の自動生成に関する研究

喜多村ら [2] は，出題者が与える解答例プログラムに対して，独自に定義したタグを用いて情報付与をおこない，解答例プログラムを解析処理し，問題を自動生成する手法を提案し，MAX/JAVA[3] という既存の学習システムで利用できることを示した．

内田 [4] は，Java 言語の空欄補充問題を自動生成するシステムである「JavaDrill」を開発した．問題番号，問題名，問題文，ソースコードをシステムに入力すると，予約語や演算子の部分に空欄を設定した空欄補充問題を自動生成する．空欄はランダムな場所に設定される．JavaDrill はプログラミング初級者を対象としており，Java 言語の文法や予約語の知識を習得させることで，プログラミング能力の育成を図ることを目的としている．

有安 [5] らは，C 言語の空欄補充問題とその解答を自動生成するシステムを開発した．本システムは，解答のプログラムコードと出題傾向を記述したデータを入力とする．出題傾向に変数，型宣言，関数，条件判定などを記入することで，問題作成者の意図に合わせて空欄を配置することができるという特徴がある．問題作成者の意図に合わせた，演習問題を大量に作成する工程の効率化を目標としている．

竹内ら [6] は，Haskell 言語の空欄補充問題を自動生成するシステムを開発した．このシステムでは，空欄補充問題の自動生成と正誤判定が可能である．また，関数や再帰構造などのプログラミング要素の重要度を考慮した効果的な穴あけ手法も提案されている．

2.2 プログラミング問題の自動採点に関する研究

Kogure ら [7] は，プログラミング学習のモニタリングシステムを開発した．このシステムでは，学習者がどのような課題に取り組んでいるかを把握することができる．学習者のプログラムのパフォーマンス情報の分析が自動的におこなわれるため，教員は学習者に対して適切な指示を与えることができるという特徴がある．

Kitaya ら [8] は，Java 言語のプログラミングの課題の提出，採点，結果返却までをウェブ上で行うシステムを開発した．プログラムの正誤判定は，コンパイラによるチェック，JUnit テスト，出力結果テストにより構成されるため，スコアリング精度が極めて高いという特徴がある．

Saikkonen ら [9] は，Scheme-robo というプログラミング演習を評価するためのシステムを開発した．Scheme-robo は，プログラムの手順，構造，実行時間を評価することができる．プログラムをアルゴリズム的な側面から分析し，プログラムの正しさを評価することができる．

また，広く利用されているオンラインジャッジシステムとして，AIZU ONLINE JUDGE[10] がある．AIZU ONLINE JUDGE は，会津大学が主催するオンラインジャッジシステムである．ウェブサイト上で公開されている問題を解き，ソースコードを提出するとその正誤が判定される．

3. プログラミング問題生成システム

3.1 概要

クラウド版 Waquema[*1] のウェブページを図 1 に示す．クラウド版のシステムでは，生成した問題を教員の Google ドライブに Google フォームとして出力することができる．「生成」を押すと，問題が生成される．「実行」を押すと，エディタに記述したプログラムを実行することができる．「設定」を押すと，使用する言語や生成する問題を選択する設定画面 (図 2) が現れる．本システムにより Google

*1 https://sites.google.com/view/waquema/

フォームとして出力された問題を図 3 に示す．出力された Google フォームには，解答と点数が設定されており，出題する問題の編集やスプレッドシートによる成績処理をおこなうことができる．対応しているプログラミング言語は，JavaScript，Python，C++である．

図 1　プログラミング問題生成システム

図 2　設定画面

4. 生成可能な問題

4.1 出力結果を答える問題

出力結果を答える問題を図 4 に示す．この問題は，問題文のプログラムを実行して得られる出力結果として適切なものを，選択肢の中から 1 つ選ぶ問題である．入力ソースコードに出力命令を含む場合にのみ生成される．選択肢は，出力結果が数値として解釈できる場合は，その数値に近い値から選択肢を作成する．出力結果がカンマで区切られている場合は，図 4 のように，カンマ区切りの異なる選択肢を生成することができる．それ以外の場合は，出力結果の部分文字列から構成される選択肢を作成する．

図 3　Google フォームとして出力された問題

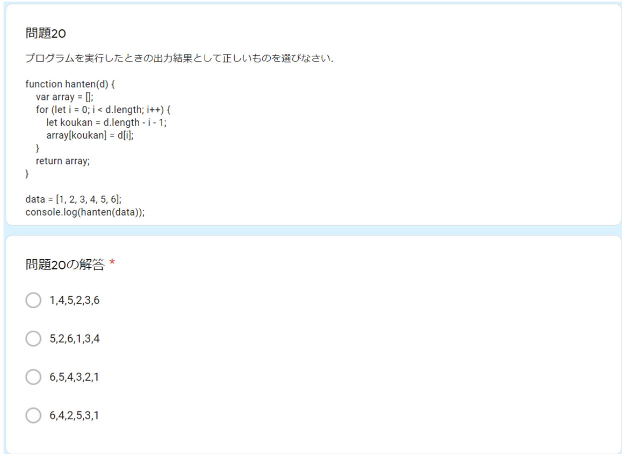


図 4　出力結果を答える問題

4.2 変数の値を答える問題

変数の値を答える問題を図 5 に示す．この問題は，問題文のプログラムにおいて，指示された地点における変数の値として適切なものを，選択肢の中から 1 つ選ぶ選択問題である．

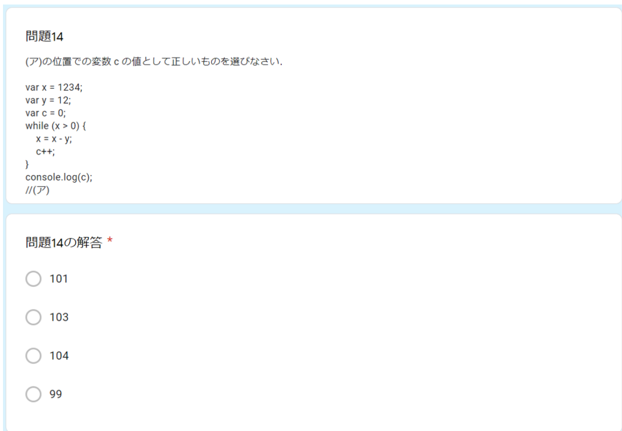


図 5　変数の値を答える問題

ⓒ 2021 Information Processing Society of Japan

4.3　変数の型を答える問題

変数の型を答える問題を図 6 に示す．この問題は，問題文のプログラムにおいて，指示された地点における変数の型として適切なものを，選択肢の中から 1 つ選ぶ選択問題である．

問題1

(ア)の位置での変数 x の型として正しいものを選びなさい．

```
var x = 1234;
var y = 12;
var c = 0;
while (x > 0) {
   x = x - y;
   c++;
}
console.log(c);
//(ア)
```

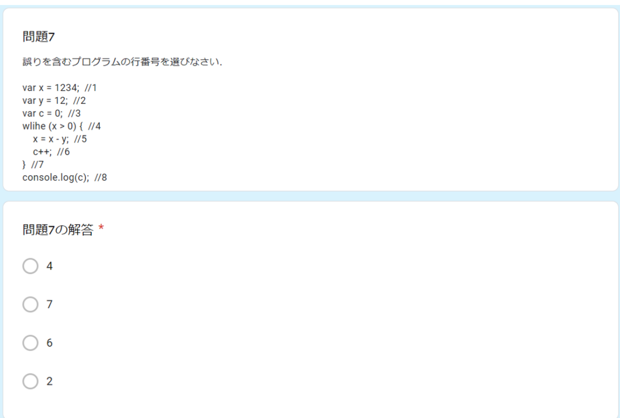
問題1の解答 *

- number
- char
- boolean
- long

図 6　変数の型を答える問題

4.4　間違い探し問題

間違い探し問題を図 7 に示す．この問題は，問題文のプログラムにおいて，スペルミスを含む行番号を選択肢の中から 1 つ選ぶ問題である．スペルミスは入力ソースコードに含まれる予約語の中からランダムに選ばれた 1 つに付与される．間違い探し問題は入力ソースコードに予約語が含まれる場合にのみ生成される．



問題7

誤りを含むプログラムの行番号を選びなさい．

```
var x = 1234;  //1
var y = 12;  //2
var c = 0;  //3
wlihe (x > 0) {  //4
   x = x - y;  //5
   c++;  //6
}  //7
console.log(c);  //8
```

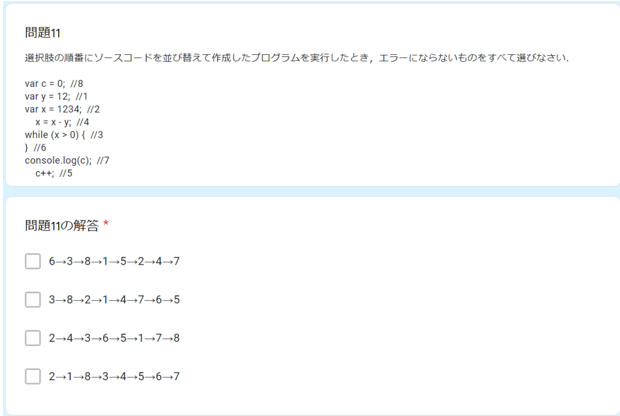
問題7の解答 *

- 4
- 7
- 6
- 2

図 7　間違い探し問題

4.5　並び替え問題

並び替え問題を図 8 に示す．この問題は，1 行ずつ分割し，シャッフルされたプログラムの断片を，選択肢の順番で並び替えたときに，エラーが発生しないものを選択肢の中からすべて選ぶ問題である．



問題11

選択肢の順番にソースコードを並び替えて作成したプログラムを実行したとき，エラーにならないものをすべて選びなさい．

```
var c = 0;  //8
var y = 12;  //1
var x = 1234;  //2
   x = x - y;  //4
while (x > 0) {  //3
}  //6
console.log(c);  //7
   c++;  //5
```

問題11の解答 *

- 6→3→8→1→5→2→4→7
- 3→8→2→1→4→7→6→5
- 2→4→3→6→5→1→7→8
- 2→1→8→3→4→5→6→7

図 8　並び替え問題

4.6　予約語選択問題

予約語選択問題を図 9 に示す．この問題は，問題文のプログラムに設定されている空欄に記述してエラーにならない予約語を，選択肢の中からすべて選ぶ問題である．

問題9

(ア)に記述して文法エラーにならないものをすべて選びなさい．

```
var x = 1234;
var y = 12;
var c = 0;
(ア) (x > 0) {
   x = x - y;
   c++;
}
console.log(c);
```

問題9の解答 *

- if
- else
- while
- this

図 9　予約語選択問題

5.　評価

5.1　学生による評価

学生には，アンケートを用いて，本システムで生成した問題を用いて作成した演習問題とテストについての評価をおこなった．集計したデータは論文のデータとして使用する予定であること，論文の執筆以外の目的には使用しないことを確認し，データの利用方針に同意できる場合のみ回答するように呼びかけた．その結果，講義を履修している学生 75 人の内，61 人から回答があった．学生に対するアンケートは 4 段階評価の設問と，複数回答が可能な設問から構成される．4 段階評価の設問の項目と結果を表 1 に示す．それぞれの選択項目は表 4 のようになっている．4 段階評価の評価値は，4 を高評価とし，1 を低評価とした．表 1 の加重平均は，表 4 の評価値を用いて，それぞれの評価値に人数を掛け，和をとった値を回答者数 $(N = 61)$ で割った値である．

5.2 理解度の確認の実施

本システムを用いて生成した問題を使用した理解度の確認を，本大学の 2 年生を対象とする JavaScript の内容を扱う講義で，2020 年 11 月 12 日から 2020 年 12 月 3 日までの期間に計 4 回実施した．各講義のデータを表 5 に示す．理解度の確認は 4 点満点であり，講義内容の理解度を確認するために実施した．学生は回答を複数回提出することができる．初回提出時の得点分布を図 10 に示す．2 回以上提出した学生の最終提出時の得点分布を図 11 に示す．各回の提出回数の内訳を図 12 に示す．

5.3 教員による評価

教員には，アンケートを用いて本システムについての評価をおこなった．集計したデータは論文のデータとして使用する予定であること，論文の執筆以外の目的には使用しないことを確認し，データの利用方針に同意できる場合のみ回答するように同意を得た上で，JavaScript 版のシステムと Python 版のシステムについて，それぞれ 1 名ずつ，合計 2 名の教員にアンケートを実施した．教員に対するアンケートは 4 択式の設問，複数選択が可能な設問，他に成できればよいと思う問題や意見を記述する設問から構成される．表 6 に複数選択が可能な設問の結果を示す．表 2 に 4 択式の設問の結果，表 3 に表 2 の設問の選択肢を示す．

6. 考察

6.1 学生による評価の考察

表 1 と表 7 についての解説をおこなう．表 1 の項目 1 についての加重平均は 3.27 であった．90%以上の学生が本システムにより生成された問題の難易度は適切であったと答えていることが分かる．また，表 7 において，70%以上の学生が自分の理解度を確認できたと答えている．本システムにより生成された問題は，授業内容の理解度を確認する上でも適切であったことが分かる．これは，教員による問題の取捨選択が適切であったためと考えられる．表 1 の項目 2 についての加重平均は 3.34 であった．90%以上の学生が「とても当てはまる」，「少し当てはまる」と答えており，「全く当てはまらない」と答えた学生は一人も見られなかった．問題の指示や解答方法は，ほとんどの学生に正しく伝わっていたと考えられる．

6.2 学生の小テストの結果の考察

表 5 についての解説をおこなう．初回提出時の得点と提出回数の間には，一貫して負の相関が見られた．初回の得点が低い学生は複数回提出していたことが読み取れる．図 11 では，最終提出時の得点が 4 点に集中している．2 回以上提出した学生は，4 点を取るまで複数回提出した学生が多くいたことが分かる．図 10 において，初回提出時に 4 点を取る学生の人数は，講義の回数を重ねる度に増加している．1 回で満点を取ることができる学生が徐々に増加しているのは，問題のパターンや解答方法を学生が理解し，小テストに慣れることができたためと考えられる．図 12 において，提出回数は，どの回の講義においても同様な分布を示している．1 つの小テストに対して，最大で 6 回提出した学生がいた．図 10，図 11，図 12 において，初回 (11/12) のデータに特徴的な分布が見られるのは，学生が講義の進め方に十分慣れていなかったことが理由と考えられる．

6.3 教員による評価の考察

表 2 についての解説をおこなう．項目 1 では，どちらの言語で本システムを使用した教員も「まあまあそう思う」と答えていた．本システムはソースコードを入力してボタンを押すだけで問題が生成できるシンプルなものであるが，「とてもそう思う」と答えた教員はいなかった．自由記述欄において，「プログラムを入力しなければ問題が生成できないというポリシーが引っかかる」，「テンプレートなどが用意されているとよい」という意見が見られた．そのため，本システムの操作の難しさは，エディタにプログラムを記述して動作確認をおこなう過程にあると考えられる．

演習問題に関する項目 2 と項目 3 に着目すると，どちらの言語においても項目 3 の方が時間を要していないことが分かる．本システムを用いたことで，Google フォームを最初から構築する時間を短縮することができたためであると考えられる．これに対して，試験問題に関する項目 5 と項目 6 では，本システムを用いた場合の方が時間を要している．演習問題の場合と異なり，どちらの言語の場合も同じかそれ以上の時間を要している．試験問題を作成する場合は，出力された問題を精査するために時間を要していたと考えられる．

項目 4 では，どちらの言語の場合も 4 割から 6 割の問題が演習問題として適切であるという回答であった．項目 7 では，JavaScript の場合は 4 割未満，Python の場合は 6 から 8 割という回答であった．JavaScript は Python に比べて文法の自由度が高いため，回答時間に制限のある試験問題としては，難易度の高い問題が多く生成されてしまうことが多い．言語間で差異が生じた理由は，言語仕様の違いにあると考えられる．

表 6 についての解説をおこなう．JavaScript の場合，演習問題では，出力結果を答える問題，変数の値を答える問題，間違い探し問題を使用したいという回答であった．試験問題では，出力結果を答える問題，変数の値を答える問題，変数の型を答える問題を使用したいという回答であった．Python の場合，演習問題と試験問題のどちらも，予約語選択問題を除く問題を使用したいという回答であった．出力結果を答える問題と変数の値を答える問題は，JavaScript と Python のどちらの言語の場合も，演習

問題と試験問題に使用したいという回答結果になった．出力結果や変数の値を問うことは，プログラミングの構造やアルゴリズムの理解度を測定する上で重要な要素であるため，どちらの言語の教員も，使用したいと答えたと考えられる．

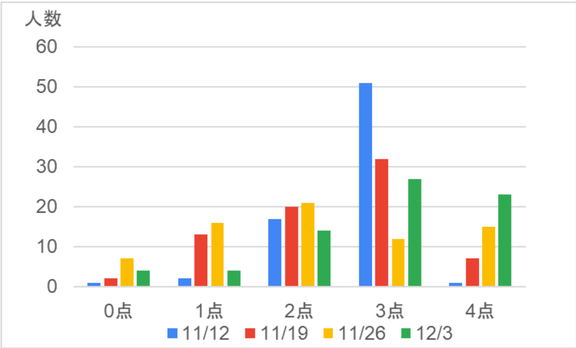


図 10　初回提出時の得点分布

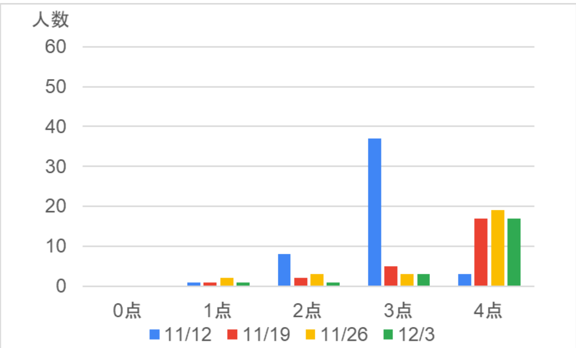


図 11　2 回以上提出した学生の最終提出時の得点分布

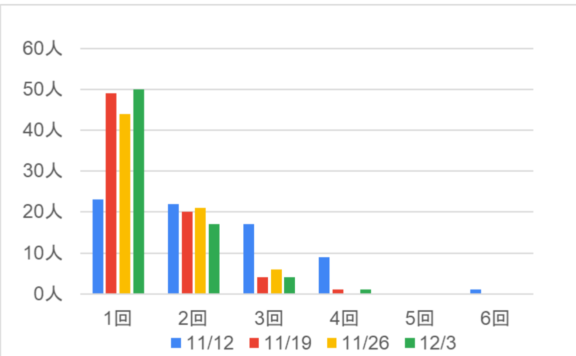


図 12　提出回数の内訳

7.　まとめ

本システムにより自動生成された問題を実際の講義で理解度の確認として使用した．学生に対してアンケートを実施し，システムの評価をおこなった．また，使用した教員へアンケートをおこない，演習課題や試験にかかる時間の評価をおこなった．学生に対するアンケートの結果からは，本システムで生成された問題の難易度と問題文の指示が適切であることが分かった．使用した教員に対するアンケートの結果からは，演習問題のような数をこなす必要がある問題を作成する場合に本システムを活用すると，時間の短縮につながる可能性があることが分かった．今後はシステムの有効な活用方法を確立するために，より多くの現場で使用してもらい，知見やデータを収集したい．

参考文献

[1] 大門巧，大西建輔．Web ブラウザ上でのプログラミング問題の生成システムの開発．情報処理学会 情報教育シンポジウム 2020，pp.75-80，2020.
[2] 喜多村 和誠，玉木久夫．タグ付き解答例プログラムからのプログラミング問題コンテンツの自動生成．情報処理学会 コンピュータと教育研究会，CE(122)，pp.1-8，2013.
[3] 山下 美穂．Web 上の Java プログラミング学習支援システム MAX/JAVA の構築，明治大学大学院理工学研究科 2012 年度修士論文，2013.
[4] 内田保雄．初級プログラミング学習のための自動作問システム．情報処理学会研究報告，CE(92)，pp.109-113，2007.
[5] 有安浩平，池田絵里，岡本辰夫，國島丈生，横田一正．学習者に合わせた C 言語演習穴埋め問題の自動生成．DEIM Forum 2009.
[6] 竹内亮太郎，大久保弘崇，粕谷英人，山本晋一郎．空欄補充問題の自動生成による Haskell プログラミング学習支援環境，情報処理学会研究報告，pp.1-8，2011.
[7] Satoru Kogure，Riki Nakamura，Kanae Makino，Koichi Yamashita，Tatsuhiro Konishi，Yukihiro Itoh. Monitoring system for the effective instruction based on the semi-automatic evaluation of programs during programming classroom lectures. Research and Practice in Technology Enhanced Learning，pp.1-12，2005.
[8] Hiroki Kitaya, Ushio Inoue. An Online Automated Scoring System for Java Programming Assignments International Journal of Information and Education Technology，Vol.6-No.4，2016.
[9] Riku Saikkonen，Lauri Malmi，Korhonen. Fully automatic assessment of programming exercises. ACM Sigcse Bulletin，ACM，33(3) pp.133-136，2001.
[10] AIZU ONLINE JUDGE，AIZU Competive Programming Club (ICPCPC)，Database Systems Lab. (online)，入手先 ⟨http://judge.u-aizu.ac.jp/onlinejudge/⟩(accessed 2021-9-1).

表 1 問題に関する評価項目と評価 ($N = 61$)

項目番号	項目	加重平均	評価値 / %			
			4	3	2	1
1	難易度は適切でしたか?	3.27	37.0	54.1	6.6	1.6
2	問題文の指示は分かりやすかったですか?	3.34	42.6	49.2	8.2	0.0

表 2 教員による評価

項目番号	項目	JavaScript	Python
1	クラウド版 Waquema の操作は簡単でしたか?	まあまあそう思う	まあまあそう思う
2	これまで，元となるソースコードから演習問題を作成し終えるには平均でどの程度の時間を要していましたか?	30 分以上	10 分～20 分
3	出力された Google フォームから演習問題を作成し終えるまでには平均でどの程度の時間を要しましたか?	20 分～30 分	10 分以内
4	演習問題として適切な問題はどの程度含まれていましたか?	4 割～6 割	4 割～6 割
5	これまで，元となるソースコードから試験問題を作成し終えるには平均でどの程度の時間を要していましたか?	30 分以上	10 分以内
6	出力された Google フォームから試験問題を作成し終えるまでには平均でどの程度の時間を要しましたか?	30 分以上	10 分～20 分
7	試験問題として適切な問題はどの程度含まれていましたか?	4 割未満	6 割～8 割

表 3 表 2 の設問の選択肢

項目番号	選択肢			
1	そう思う	まあまあそう思う	あまりそう思わない	そう思わない
2，3，5，6	10 分以内	10 分～20 分	20 分～30 分	30 分以上
4，7	8 割以上	6 割～8 割	4 割～6 割	4 割未満

表 4 表 1 の評価値の設定

項目	評価値
とても当てはまる	4
少し当てはまる	3
あまり当てはまらない	2
全く当てはまらない	1

表 5 講義ごとのデータ

項目	11/12	11/19	11/26	12/3
出席者数	72	74	71	72
平均点	2.68	2.39	2.16	2.84
平均提出回数	2.22	1.41	1.46	1.38
初回提出時の得点と提出回数の相関係数	−0.13	−0.38	−0.37	−0.26

表 6 問題に関する複数選択式アンケート

どの種類の問題を使いたいと思いましたか?				
項目	JavaScript		Python	
	演習問題	試験問題	演習問題	試験問題
出力結果を答える問題	○	○	○	○
変数の値を答える問題	○	○	○	○
変数の型を答える問題	×	○	○	○
並び替え問題	×	×	○	○
間違い探し問題	○	×	○	○
予約語選択問題	×	×	×	×

表 7 複数選択式アンケートの質問と結果 ($N = 61$)

問題を解答してどのように感じましたか?	
項目	割合
自分の理解度を把握できた	73.8%
難しいと感じた	31.1%
先生が手作業で作った問題だと感じた	14.8%
出題された問題は復習に向いていないと感じた	6.6%