

Study on Computer-Adaptive Testing: Proposal of a Scaffolding Tool

Hiroataka Azuma, Tokyo Denki University, Japan
Hironobu Takenouchi, Tokyo Denki University, Japan
Tatsuyuki Takano, Kanto Gakuin University, Japan
Osamu Miyakawa, Tokyo Denki University, Japan
Takashi Kohama, Tokyo Denki University, Japan

The Asian Conference on Education 2020
Official Conference Proceedings

Abstract

A multiple-choice method is one of the major methods in Classical Testing Theory used at university, in certification exams and so on. This method enables faculty to score a test easily while requiring a certain amount of quizzes to estimate examinees' abilities with a certain accuracy. This also requires time to answer for examinees and time to create quizzes for faculty. Then it can be considered to adopt CAT, Computer-Adaptive Testing. CAT is another testing theory that reduces the number of quizzes to theoretically half compared to the existing multiple-choice method while keeping the accuracy of estimating examinees' abilities almost the same; however, it is necessary for each quiz to have its own difficulty. In this study, a programming exercise tool with a scaffolding method is going to be suggested aiming at generating quizzes with various difficulties automatically. A scaffolding method is a method that helps examinees to solve problems by providing some hints gradually. By using this tool, what elements are associated with reduction in difficulty can be examined, and they will be used to a quiz generation. Students will be required to answer half the number of quizzes and faculty will need to create much less ones than before. This tool also supports beginners to avoid bias in a difficulty distribution by providing very basic quizzes so that the beginners will be encouraged to brush up their skills and more detailed classification will be acquired.

Keywords: Computer-Adaptive Testing, Item Response Theory, Scaffolding

iafor

The International Academic Forum
www.iafor.org

Introduction

At university and in certification exams, there are a lot of testing methods to estimate examinees' abilities. One of the most popular ones is a multiple choice. A multiple choice provides some options for an examinee and let him/her to choose one of them. By using this method, it is easy for faculty to score a test. But to estimate examinees' abilities correctly, it is necessary to create a lot of quizzes in a single test. That means this method takes a long time for examinees to answer and also for faculty to create quizzes. Since these reasons, it can be thought to adopt a CAT.

CAT stands for Computer-Adaptive Testing. CAT is one of the computer-based testing methods. By adopting CAT, both faculty and examinees can enjoy the fruit of its traits. First, quizzes given to examinees vary according to individuals. Second, the number of quizzes also varies. Because of these reasons, CAT can contribute to preventing cheatings and decreasing the time to answer quizzes theoretically to half compared to the multiple choice. But examinees cannot go back to the previous quizzes since CAT decides which quiz should be next based on the examinee's answer to the previous one. And most importantly, each quiz in a test must have its own difficulty beforehand. Thus before using CAT, a lot of quizzes are necessary, the difficulties of which are well distributed.

Our study has gone through a plenty of experiments to calculate the difficulty of a quiz and to create a hard one by using some source code metrics or calculating coefficients. And some of them have been found to be related to the difficulty. But nothing can prove that it is truly effective. Because of these, this study's way has been changed slightly. This study's hypothesis is that if some hints are given, the difficulty of a quiz goes down to a certain degree, in other words, hints make a quiz easier. Thus quizzes with various difficulties will be gained.

Since minute changes to the difficulty of a quiz are hoped to be obtained, a scaffolding method has been introduced. A scaffolding method is a method to support examinees to achieve their goals that they cannot solve by themselves alone. This study proposes a tool that gives hints gradually.

Related Studies

Wood says the roles of instructions including "scaffolding" are getting attentions, limiting the options, keeping motivations, highlighting points, managing frustrations, and doing demonstrations (Wood, 1976). Getting attentions is to orient an examinee's attention towards a task to let him/her concentrate on it. Limiting the options is for reducing the options so that an examinee can focus only on necessary things. Keeping motivations is to maintain an examinee's motivation on a task for his/her stable concentration. Highlighting points means showing some points are related to each other to give further information to an examinee. Managing frustrations is for avoiding an examinee's strong dependence on an instructor. Lastly, doing demonstrations means showing an ideal result or way to solve and let an examinee do in the same way.

Collins says "scaffolding" is included in classical apprentice system (Collins, 1991). In classical apprentice system, trainees (in this case, examinees) used to observe what

their instructors do, and to try to learn. But today, modern apprentice system has been changing that instructors support their trainees to achieve some tasks. And their helps gradually decrease as the trainees are getting skilled.

Stone says “scaffolding” can be divided into four steps: Firstly, orienting an examinee’s attention to a task where the difficulty of it is a little bit harder compared to the examinee’s ability. Secondly, deciding how helps are necessary for an examinee to solve a problem. Thirdly, instructors should use various ways of helps according to a task. Lastly, supports should be decreased as an examinee gets skilled (Stone, 1998).

Ueno proposes a system that uses a scaffolding method for programming learners (Ueno, 2015). Ueno suggests that the learning effect becomes the highest when a hint that enables an examinee to answer it correctly almost fifty percent is given.

A Scaffolding Tool

In this study, a scaffolding tool provides a platform where examinees can train their programming skills. This tool gives a quiz, then gives the next one that is chosen to fit with the examinee’s ability adaptively based on his/her answer to the previous one. More concretely, this tool constructs of GUI and background functionality showing a source code to solve, left time to answer, statement texts to make sure how to answer the quiz, an answer column to type in, and further information.

In order to estimate an examinee’s ability and a quiz’s difficulty, this tool uses IRT to achieve them. IRT stands for Item Response Theory. In IRT, it is assumed that each examinee has his/her own ability that is hidden and independent from each other. IRT can calculate an item (in this case, a quiz)’s difficulty independently from statistical population, and also calculate an examinee’s ability as well. IRT provides three models to estimate. In this study, two-parameter model that uses an examinee’s ability, an item’s difficulty and discrimination factor has been adopted. This theory is used for various ways such as analysis (Nakashima, 2017) and assessment (Tsutsumi, 2018).

This tool also provides some hints to implement a scaffolding method. Steps of the hints are: Step1, no hints are given. The tool gives only a source code and statements. Step2, trace information is given. This means the tool highlights a line for each step of execution to show which line works at each time. Step3, trace and variables information are given. In addition to the step2, it shows variables’ information so that an examinee can know what each variable contains. Step4, trace, variable and standard output information. In addition to the step3, it shows what outputs will be shown in a console except last one.

This scaffolding tool runs as a following flow. It shows a source code and statements to let an examinee to solve with no hints. When the examinee answers the quiz correctly, the next quiz that is a little bit harder will be shown. When answering wrongly and not the all hints are given, the next hint will be provided. When answering wrongly and all the hints are already given, the next quiz that is a little bit easier will be shown. When the fluctuation of estimating the examinee’s ability converges, the test ends.

The GUI of this tool is shown in Figure 1. The largest area is for showing a source code. Under the source code area, there are the statement area and the answer column. The button written as “Send” is to send the answer when the examinee pushes this. On the right-hand side, there are an area to show left time to answer, hint status and a trace button (but the last two are hidden because the test has not begun yet), the variable-information area and the standard output area. An examinee can choose a test file selecting “Open” (it’s hidden in this figure) menu item in the “File” item in the menu bar on the upper area and can change the language in the “Settings” item.

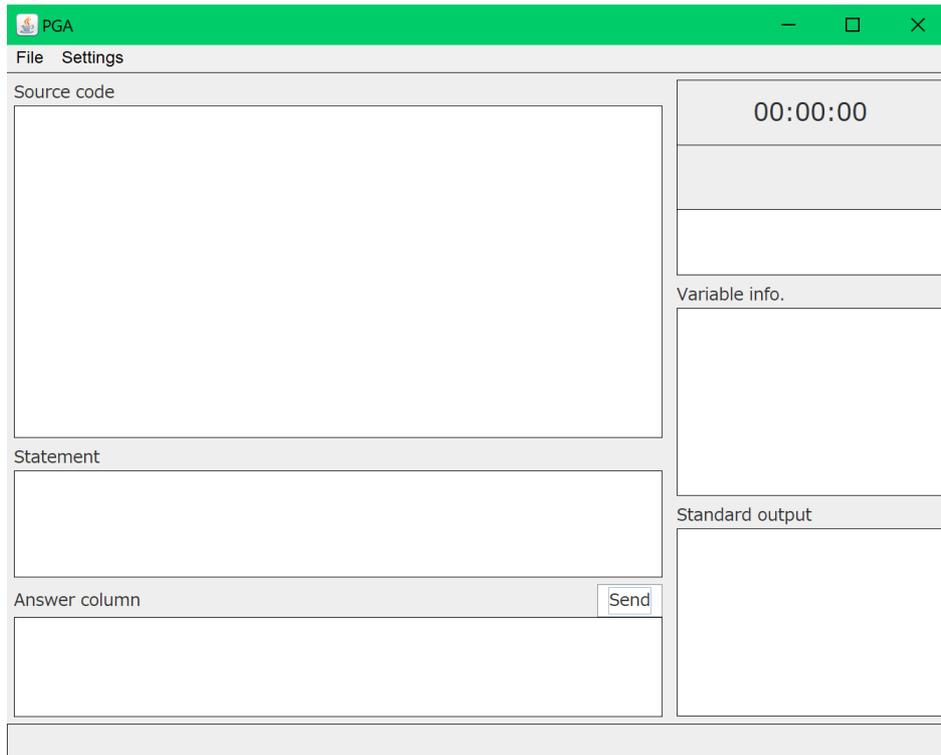


Figure 1: GUI of the scaffolding tool

When the test begins, each area works as shown in Figure 2. The source code is being shown with leading numbers. The statements are also shown. And the left time begins ticking. The hint status indicates what hints are available saying “No hints.” The trace button also activates (but in this case, it is disabled and saying “No hints given” since there are no hints available).

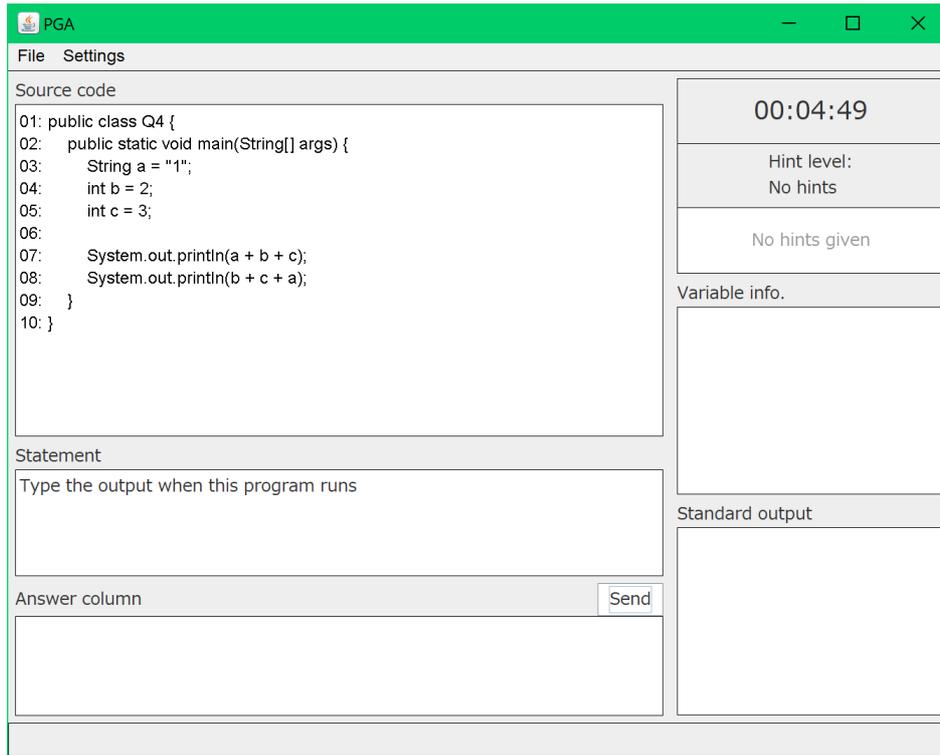


Figure 2: GUI when the test has begun

Each examinee is required to answer the correct output in console when a program with the source code runs. This is a reason why this study chooses two-parameter model in IRT since three-parameter model needs guessing information (this is mostly used in such as a multiple choice quiz) while this type of answers (free description) gives very low probability for examinees to answer it correctly when he/her type in by guessing.

After an examinee answered wrongly, the tool updates its hints. The trace information hint provides a highlighted line. The examinee can proceed the line by pushing the trace button that says “Click here to the next line” line by line. Regardless of the repetition, branching and method calls, the whole executions will be shown. GUI of this case is shown in Figure 3.

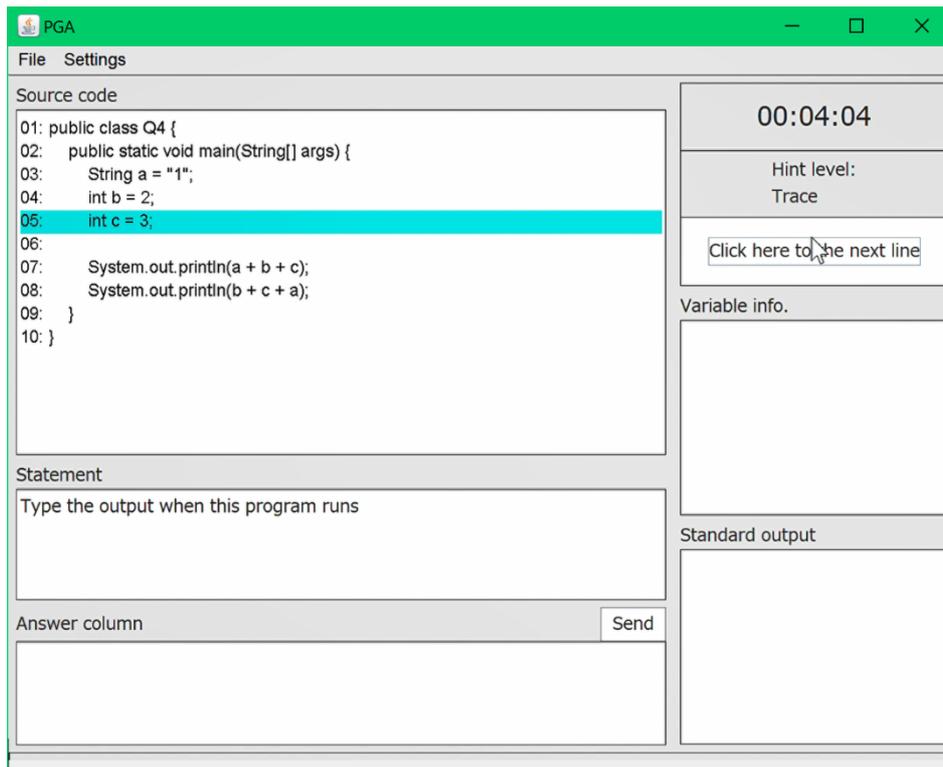


Figure 3: GUI when the trace information is given

When an examinee answers it wrongly even after the first hint was given, the tool updates its hints again as shown in Figure 4.

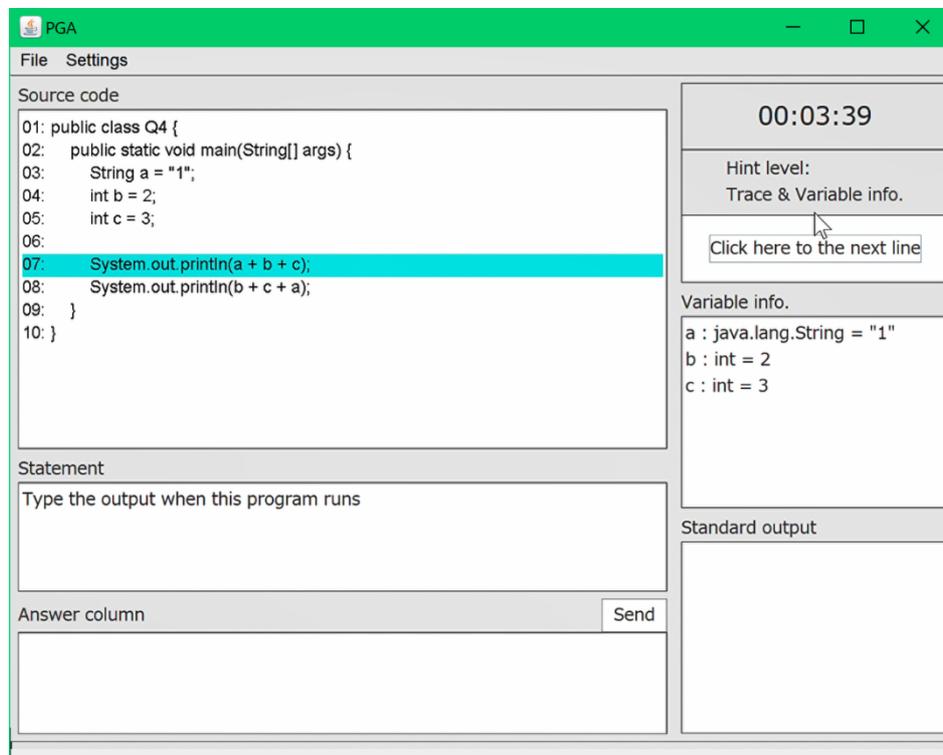


Figure 4: GUI when the variable information is given

In this step, variables information is provided with the types and values contained in them until the line executed at that time. Even if the line is in another method, it shows correct information showing only variables available at that time.

When an examinee reaches the last step of hints, the standard output information is given to see as shown in Figure 5. Thus the examinee can know which line shows what output in console. But the last output line is not be shown because the whole outputs are the quiz's answer themselves. If the source code contains only a single line to show output, the standard output area gives no information. This is the last step of the hints. So when the examinee answers it wrongly even if all the hints were provided, the tool provides the next, a little bit easier quiz.

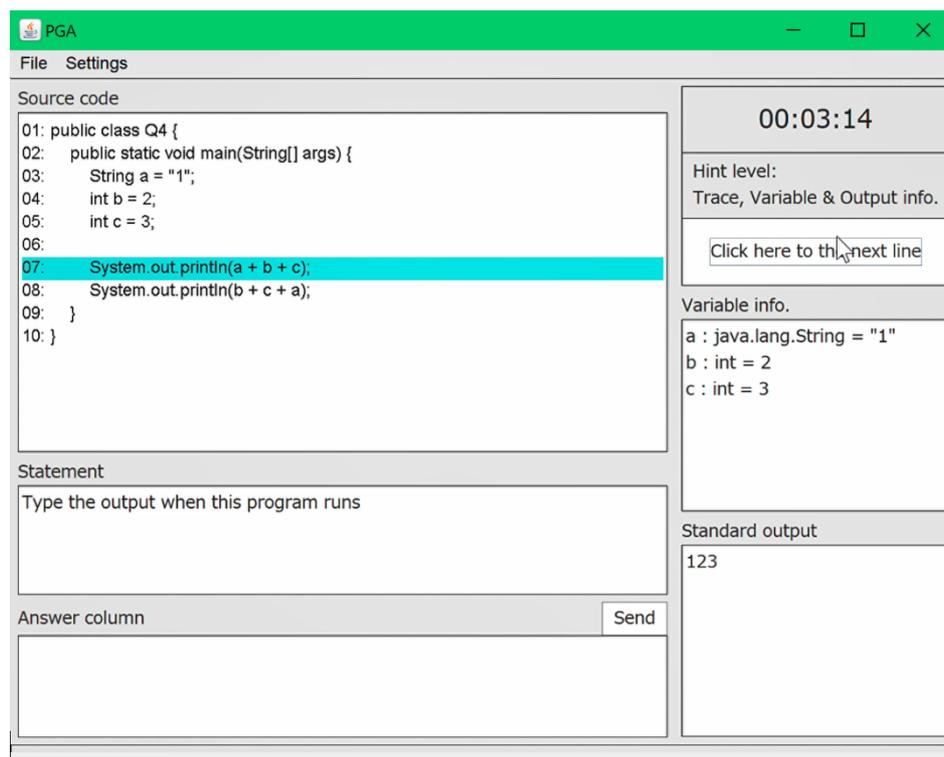


Figure 5: GUI when the standard output information is given

Through these gradual hints, the tool estimates an examinee's ability and the difficulty of a quiz to show the next one adaptively.

Additional Tool for True Beginners

This study assumes that true beginners may not solve easiest quizzes that their instructors prepare. This is actually an important factor to avoid because their answers may cause raising the average of the difficulty of each quiz.

To prevent this problem, an additional tool is provided in this study. This additional one provides quizzes that mainly consist of very basic knowledge.

This additional tool is implemented as Web service. The UI of the tool is shown in Figure 6.

次の選択肢①、②から、「インデントが誤っているソースコード」を選択せよ。

選択肢

①

```
public class ClassName {
^  public static void main(String[] args) {
^  ^  for(int i = 0; i < args.length; i++) {
^  ^  System.out.println(args[i]);
^  ^  }
^  }
}
```

②

```
public class ClassName {
^  public static void main(String[] args) {
^  ^  for(int i = 0; i < args.length; i++) {
^  ^  ^  System.out.println(args[i]);
^  ^  }
^  }
}
```

Figure 6: UI of the additional tool

The statement on the top says “Choose the source code that contains wrong indentation from 1 and 2 below.” Proper indentation is of grave importance, but the scaffolding tool assumes that all the examinees have already known it. Thus this additional tool supports the basics.

Another example of the additional tool is shown in Figure 7. The statement says “Choose the source code where grayed areas indicate the type of parameters from 1 and 2 below.” Ability to decide which is a type of parameter is also a very important factor.

次の選択肢①、②から、「**網掛け部分が仮引数の型**であるソースコード」を選択せよ。

選択肢

①

```
public class ClassName {  
    public static void main(String[] args) {  
    }  
    public static String getExamInformation(int score) {  
        return "";  
    }  
}
```

②

```
public class ClassName {  
    public static void main(String[] args) {  
    }  
    public static String getExamInformation(int score) {  
        return "";  
    }  
}
```

Figure 7: Another example of the additional tool

Conclusion

A scaffolding-included tool for programming learners is introduced to obtain quizzes with various difficulties. This study aims for gathering variety of quizzes automatically, and for being part of functionality in CAT system.

This study focuses on not only a source code itself but also other pieces of information such as statements and hints. Our past experiments tried to create a difficult source code without any other factors. But this study considers a quiz, which contains various factors, so that an easier one can be generated by providing some hints.

References

- Wood, David, Jerome S. Bruner, and Gail Ross. (1976). The role of tutoring in problem solving. *Journal of child psychology and psychiatry* 17.2, pp.89-100.
- Collins, Allan. (1991). A Cognitive Apprenticeship for Disadvantaged Students.
- Stone, C. Addison. (1998) The metaphor of scaffolding: Its utility for the field of learning disabilities. *Journal of learning disabilities* 31.4, pp.344-364.
- Maomi Ueno, Junya Matsuo. (2015). Scaffolding System Provides Adaptive Hints Using Item Response Theory. *The Journal of the Institute of Electronics, Information and Communication Engineers D98.1*, pp.17-29.
- Yasuhiro Nakashima, Yoshikatsu Sasaki. (2017). Koumokuoutouriron ni yoru suugakukisoryoku kakunintesuto no kaiseki [Analysis of Tests that Confirm Basic Mathematic Ability by using Item Response Theory]. *Bulletin of Kurume Institute of Technology* 39, pp63-71.
- Emiko Tsutsumi, Masaki Uto, and Maomi Ueno. (2018). Dainamikku asesument no tameno kakure marukohu irt moderu [Hidden Markov IRT model for Dynamic Assessment]. *The Japanese Society for Artificial Intelligence* 32.

Contact email: 19JKM02@ms.dendai.ac.jp