

Automatic Grading System to Supporting Blended Learning in Basic Programming Practice – an Experience Report

Ani Rahmani¹ and Joe Lian Min²

^{1,2} Computer Engineering and Informatics Department

^{1,2} Bandung State Polytechnic (Politeknik Negeri Bandung – POLBAN)

^{1,2} Bandung, West Java, Indonesia

¹anirahma@jtk.polban.ac.id, ²joelianmin@jtk.polban.ac.id

Abstract—The problems in teaching programming are widely known. Research on teaching programming has been carried out both in theoretical and practical. Especially in the practical, the point emphasized that students must practice consistently and discipline. The problem is that not all programming tasks can be assessed because it takes too long. This situation will cause students to be apathetic or stop programming. An automatic grading (AG) in teaching programming practices is an alternative to overcome this problem. AG will check the assessment program directly. Integrating AG with the Learning Management System (LMS) helps teachers to evaluate. The use of AG in Blended Learning (BL) is implemented in five activities: profiling students, determining the material, designing questions, assessment-evaluation, and providing feedback. The effectiveness of BL can be seen in two ways: quantitative and qualitative. Quantitatively, students who do the exercise continuously and discipline, have right in skills and final evaluation results. Qualitatively more than 90% students state that they are enjoying the practice class, motivated to finish their tasks, and greatly helped. This article is an experience report of the use of AG in applying BL for teaching programming practices.

Keywords: *blended learning, automatic grading, teaching programming practice, feedback, assessment and evaluation*

I. INTRODUCTION

Research on teaching programming has been carried out with a various perspective both educational and industry, especially software industry. It can be understood because programming is an intellectual activity that is quite complex. It is also considering the ‘must be’ possessed skill by first-year students in Information Technology or Informatics department [1]. In order to achieve good practically programming skill, programming practices need to be carried out continuous, consistent, and discipline.

The problem in teaching programming practice is the teacher difficulty in checking all student programs because it takes too long. The number of programs needed to be check to increase along with running semester. The most unexpected case is it cause students become apathetic and even more stop hone their programming skill.

To overcome this situation, automatic grading becomes one of the alternative solutions to support BL In the teaching programming context. The BL is playing the role to examine

students' programming tasks, manage the assignments, give fast feedback, and carry out the assessments and the evaluations.

This article is an experience report of the use of automatic grading in the application of BL, in teaching basic programming practices at Bandung State Polytechnic - D4 Program, with observation objects taken from teaching in 2014 until 2018. The automatic grading technology was used the Online Judge integrated with model as LMS. Besides explaining the technical aspects of applying the automatic grading, the discussion also raised the benefits obtained from the implementation of the BL.

II. LITERATURE REVIEW

2.1 Teaching Programming

In general, teaching in programming is divided into two parts: teach the theory and do the practice. Teaching the theory emphasizes knowing the terminology, understanding the basic concepts, enhancing their logic, and synthesizing the theory. While the practice helps the students to sharp their skills, enhancing their motoric skills, and practice programming [2]. The success indicator learning on programming depends on many aspects includes: the number of solved problems, thinking in a logical way, and the number of solved high-level problems [3].

The problems faced by students on programming includes [4]:

1. Understanding the assignments,
2. Solving the problem on programming perspective,
3. Designing the solutions for the problems given,
4. Expressing the solutions into a program,
5. Divide the problem into certain functions (modular),
6. Transforming the solutions, from human languages into programming languages,
7. Utilizing the experience of previous assignments to design solutions for the next task,
8. Tracing the programs to ensure the solution compatibility with the problems.

2.2 Automatic grading system

Automatic grading is a web-based assessment application, act as a receiver of the submitted source code on the programming task [5]. The program is compiled and tested, then checked according to the environment used. Currently automatic grading is more popular, and it used in competitive programming, enhancing education and recruitment processes, facilitating the solving of data mining challenges, online compilers and development platforms as integrated components of other custom systems [6]. The working principle of an automatic grading is shown in Figure 1.

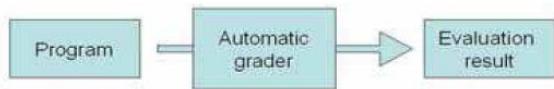


Figure 1. How the Automatic Grading System Works [7]

There are many automatic grading software that can be free accessed (open source) with variation in abilities. Some automatic grading works on particular programming languages and others works on several programming languages. One of the automatic grading developed in Indonesia was initiated by the Bandung Institute of Technology (ITB), through a long evolution. The evolution of the system is illustrated in Figure 2.

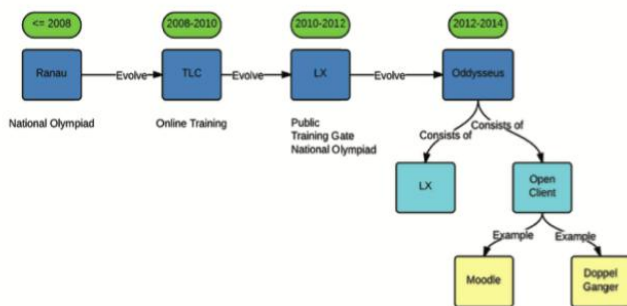


Figure 2. Evolution of automatic grading maintained by ITB [8]

Larges number of automatic grading systems that can be used free. One of them is Online Judge plug-in of Moodle. Online Judge can be download from https://github.com/hit-moodle/moodle-local_onlinejudge/ and must be installed on Linux environment (preferably Ubuntu or Debian).

In general, automatic grading applications assess three aspects of a program [9]:

- Assess the validity of the source code (check the output of the source code),
- The completeness of the source code,
- The quality and indentation of source code.

For teaching needs, the value obtained by each student in each exercise or assignment needs to be managed by the teacher for evaluation needs. By utilizing Online Judge teachers can assess and evaluate the students' task easily.

2.3 Automatic grading in a blended learning system

BL is achieved by combining various modes of delivery, teaching models, and student learning styles effectively [10]. While [11] shows that BL includes 5 things:

- Synchronized learning and allows all students to participate at the same time, led by the teacher.
- Provision of learning materials that can be accessed by students at any time
- Collaboration, where students can communicate and be creative with others, via e-mail or discussion via chat.
- Assessment, which is a measure of students' abilities and knowledge. The assessment process can be carried out before the learning process and the next is reassessed in the middle and after the learning process. Assessment is not limited to conventional tests, quizzes but also includes narrative feedback, portfolio evaluation and, more importantly, a teacher's reflection on the effectiveness of BL. Assessment is one of the most important elements of BL, for two reasons, namely: a) allowing students to test the content they already know, to measure their own experience and learning abilities; b) Measuring the effectiveness of all learning activities
- Supporting Materials: These include both physical and virtual reference materials to improve retention and learning, can be printed references, work aids, and personal digital assistants.

The success of the application of BL depends on many variables, such as the quality of the material, the ability of students in an interactive environment with technology, the ease of operating supporting technology, and the learning environment. BL is also influenced in large part by student attitudes towards the assignment given [10]. BL is mainly implemented in a virtual learning environment (VLE). The main objective of BL is to create a richer learning process, with students being motivated by combining face-to-face sessions with appropriate online activities [12].

Furthermore, in the learning process, another important thing is an assessment and giving fast feedback. Assessment can be a reference for students to measure their abilities. It also can be a source of providing feedback for students and teachers. For effective learning, students need to measure their abilities on a task, compared to what they have to do to close the gap with other students [13].

III. OBSERVATION AND EXPERIENCES

There are five activities in applying BL using automatic grading: a) profiling students; b) determination of material that can be assessed automatically; c) designing questions or programming tasks; evaluation and evaluation, and giving feedback.

3.1 Profiling student

A student profiling is done a few weeks before the teaching begins. The technical implementation of profiling is to spread questionnaires to new students, to find out the experience, knowledge and programming activities that they already have before they take part in basic programming

lectures. It is important activity to find out the diversity of students' abilities in programming.

Some students have known programming since junior high school (SMP). In addition, students graduating from the Vocational School (SMK) in software engineering, have mostly been involved in a number of software development projects, both in the form of school assignments, as well as in workplace projects (many vocational students work as part timers). Another phenomenon is that a number of students from high school (SMA) have experienced in a number of programming competitions, both at the regional and national levels.

The diversity of students' abilities needs to be addressed with certain teaching strategies. Therefore, students who are just starting to learn can take part in learning, and experienced students can still enjoy the learning process. It is not expected that the diversity of students' abilities in the classroom, does not makes the classroom atmosphere conducive.

Automatic grading in BL supports teaching strategies to deal with a diversity of knowledge and skills. Practice questions are designed in such a way and gradually in accordance with the syllabus. On the other hand, practice questions are also made with higher complexity, for other students who expect more challenging questions.

3.2 Determination of material scope

Basic programming material generally starts from the principles of programming, data concepts, process structures (sequential, branching, and loops), module, and concept of files. Due to the limited ability of the Online Judge system to evaluate a program, the file material cannot be assessed automatically, so that the examination and assessment of the material is done manually.

The questions for all programming practice teaching materials are designed into 3 schemes:

Reading a program, for example it is realized that students must complete the blank part in the program, or correct the provided program.

Modifying the program. For example, in this case, students are required to change the given program to obtain the certain value.

Building a program. This is the challenging part of programming case. Students have to build the entire answer started by looking for a design to implementation it into a C programming language because the Online Judge plugin Moodle can only check and assess the C programming language.

3.3 Designing programming assignment

There are certain rules have to consider in designing a question to be checked and assessed by an automatic grading system. Technically, the rules must be understood by a designer and the strategy to evaluate them. Figure 3 is an example of a problem by following rules, which required to explain the expected input and output specifications as well as examples.

Selisih Prima

Nama file: `SelisihPrima_Nama.c`

Dalam matematika, bilangan prima adalah bilangan asli yang lebih besar dari angka 1, yang faktor pembaginya adalah 1 dan bilangan itu sendiri.

Diketahui bilangan X dan Y, hitunglah selisih bilangan prima terbesar dan bilangan prima terkecil mulai dari X sampai Y.

Spesifikasi Input

Satu baris berisi dua buah bilangan positif X dan Y ($2 \leq X < Y \leq 32.750$). Pada jangkauan dari X sampai Y, dijamin ada minimal satu buah bilangan prima.

Spesifikasi Output

Satu baris berisi bilangan bulat yang menyatakan selisih bilangan prima terbesar dan bilangan prima terkecil mulai dari X sampai Y.

Contoh Input

2 10

Contoh Output

5

Figure 3. Writing programming assignment in the Online Judge

There is an assessment strategy for every question by determining the number of tests, the weight of the value of each test, and the input-output value of the program for each test. Figure 4 shows the number of tests performed on the question in Figure 3, while Figure 5 is the determination of the weight of values and input-output on a test. Figure 5 is an assessment form, where one of the assessment processes weighs 5% with the input value of the program entered is 3 37 respectively, and the output must be 34.

- ▶ Test Cases1 ?
- ▶ Test Cases2 ?
- ▶ Test Cases3 ?
- ▶ Test Cases4 ?
- ▶ Test Cases5 ?
- ▶ Test Cases6 ?

Figure 4. Setting the number of test case in a program

▼ Test Cases3 ?

Grade ? 5%

Testcase from ☐ files

Input ? 3 37

Output ? 34

Figure 5. Designing the test case and assessment on a program

All assignment of programming should be validated before use, by preparing answers in the form of source code, checking and evaluating by automatic grading. All questions must be guaranteed to be valid and they have been accepted in all tests.

3.4 Assessment and Evaluation

Programs that are checked and assessed by automatic grading produce values within a range of 0-100. Figure 6 shows that the assessment score is 100, because all tests are accepted. In Figure 7, the score of the assessment does not reach 100 because the assessed program is only correct in 3 of the 6 tests, which every test has a certain weight.

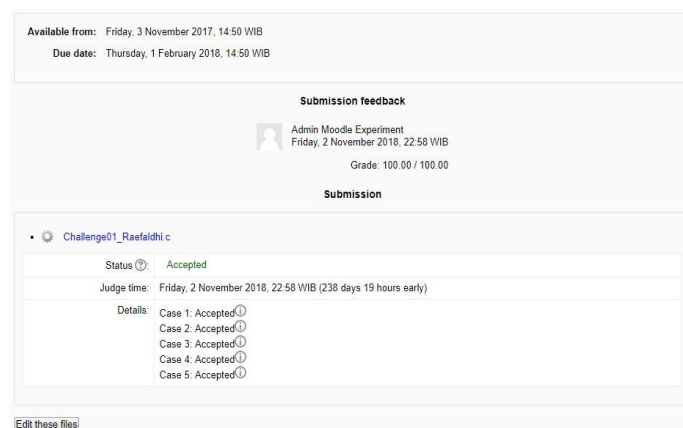


Figure 6. The result of assessment of a program

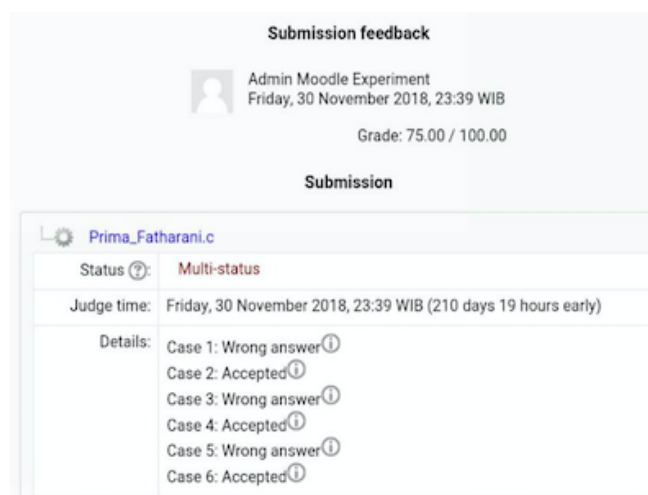


Figure 7. The result of assessment only 3 accepted

In the teacher's side, the scores of the assessment can be seen through Moodle. All student grades in every exercise or assignment are collected in a downloadable sheet, then it can be exported into Ms. Excel format. The result can be presented visually using charts in Ms. Excel or graphics in Moodle. Figure 8 shows the results of automatic grading from the instructor's side for 1 assignment, and Figure 9 shows the results of the assessment for all students in all assignments.

First name / Surname	ID number	Institution	Grade	Comment	Last modified (Submission)	Last modified (Grade)	Status	Final grade
Alvira Darajat	181524002	Politeknik Negeri Bandung	100 / 100	Accepted SelishPrima_vira.c	Thursday, 7 February 2019, 17:54 WIB	Thursday, 7 February 2019, 17:55 WIB	Update	100.00
Chofef Endamara	181524004	Politeknik Negeri Bandung	100 / 100	Accepted SelishPrima_Chofef.c	Thursday, 20 December 2018, 18:17 WIB	Thursday, 20 December 2018, 18:17 WIB	Update	100.00
Hafiz Fadli	181524010	Politeknik Negeri Bandung	100 / 100	Accepted SelishPrima_Hafiz.c	Monday, 3 December 2018, 14:14 WIB	Monday, 3 December 2018, 14:14 WIB	Update	100.00
Ananda Fauzan	181524003	Politeknik Negeri Bandung	100 / 100	Accepted SelishPrima_Ananda_Bayu_Fauzan.c	Monday, 5 November 2018, 21:12 WIB	Monday, 5 November 2018, 21:12 WIB	Update	100.00
Raefaldhi Junior	181524026	Politeknik Negeri Bandung	100 / 100	Accepted SelishPrima_Raefaldhi.c	Saturday, 3 November 2018, 07:25 WIB	Saturday, 3 November 2018, 07:25 WIB	Update	100.00

Figure 8. The automatic grading result on an assignment from the teacher's side (Moodle)

First name	ID number	Institution	Lathian 1	Lathian 2	Lathian 3	Lathian 4	Lathian 5	Lathian 6
Muhammad Alkhalagi	181524021	Politeknik Negeri Bandung	10.00	10.00	10.00	10.00	10.00	10.00
Evan Lukayya	181524007	Politeknik Negeri Bandung	10.00	10.00	10.00	10.00	9.33	9.00
Raefaldhi Junior	181524026	Politeknik Negeri Bandung	10.00	10.00	10.00	10.00	8.67	10.00
Fatharani	181524009	Politeknik Negeri Bandung	10.00	10.00	10.00	10.00	10.00	10.00
Rayhan Putra	181524028	Politeknik Negeri Bandung	10.00	10.00	10.00	9.33	8.00	10.00
Muhammad Fadli	181524020	Politeknik Negeri Bandung	10.00	10.00	9.38	8.00	7.33	-
Devianda Suhanda	181524006	Politeknik Negeri Bandung	10.00	8.00	10.00	10.00	10.00	8.00
Ananda Fauzan	181524003	Politeknik Negeri Bandung	10.00	10.00	8.13	9.33	9.33	10.00
Riyansani Putra	181524029	Politeknik Negeri Bandung	10.00	10.00	8.75	9.33	9.33	10.00
Rahman Riyad	181524027	Politeknik Negeri Bandung	10.00	10.00	10.00	9.33	10.00	10.00
Zafan Arifiansang	181524032	Politeknik Negeri Bandung	10.00	10.00	9.38	8.00	9.33	10.00
Roma Pasaribu	181524030	Politeknik Negeri Bandung	8.00	10.00	8.75	8.67	8.00	8.00
Jafer Shadiq	171524014	Politeknik Negeri Bandung	8.00	10.00	10.00	9.33	9.33	9.00

Figure 9. The result of automatic grading assessment of all students in all assignments (Moodle)

Student progress in learning can be observed via scoring on Moodle every week or when needed can be downloaded. Teachers can monitor students who cannot achieve the target of assignment. The potential problems are:

Students do not do the given assignments. For this case, students need to be confirmed immediately for the reason. Teachers should be wise to know students' problems, because sometimes students have technical issues in submitting the assignment, which one of those is due to internet connection trouble.

Students who are really not doing assignment, teacher can give certain intervention steps, that perhaps students realize the importance of experience doing programming exercise in learning process.

Some of students' programs are still not perfect yet, so not all the testing cases are accepted. For this case, students require feedbacks why the test is not accepted. Teachers can give feedbacks via chat or email on Moodle.

Although the students succeed the assignment, it is also necessary that teachers download the source codes to check their structure and justify whether the source codes meet the clean code perspective or not. This is important to emphasize students to write programs with well-structured scholars and pay attention on a good program writing procedures.

3.5 Giving fast feedback

Teachers can do analysis and evaluation of all student achievements in learning with an automated assessment process. Thus, giving feedback can be delivered immediately and the teachers can determine the strategy to improve

teaching process. On the other hand, quick feedback makes students feeling seriously cared for, appreciated, and ultimately motivated. Figure 10 and Figure 11 shows some examples of giving feedback to students through chat or email using Moodle.



Figure 10. The feedback on writing programs by following certain standards.



Figure 11. The feedback on understanding domain problem.

IV. DISCUSSION

Analysis of implementation of automatic grading in BL is conducted by quantitative and qualitative.

4.1 Quantitative analysis

Quantitative. The observations are done by doing training activities and programming tasks, which include evaluation results, daily evaluations in the form of quizzes as well as evaluations in the middle and end of semester.

From the existing data, the teachers can carry out various class actions and various teaching strategies to improve the quality of the learning process. If needed, the teachers communicate with the instructor of other subjects to validate the findings. The right action can be carried out in the class if it is supported by valid and up to date data.

4.2 Qualitative analysis

Qualitative analysis is to see students' perceptions in enjoyment aspect, motivation, and continuity in programming practice. Then, it is associated with the understanding level and students' skills in each material.

There is a fact that the perceptions of students with teachers' perceptions can be different. There are students who feel not understand, even though in the viewpoint of the teachers that the students understand. They still need to practice a lot. On the other hand, there are students who feel understand, but it contradicts to teachers' point. This fact becomes a source for teachers to take certain actions, both for all students, and for certain individuals.

Nevertheless, there is an obstacle to know originality of students' programs when using automatic grading because programming activities is done without any supervisions. Therefore, it is high possibility if some students doing unfairly in completing assignment such as cheating or doing plagiarism from other sources. This automatic grading system is only used for routine programming practice to support formal learning. For instance, teachers design quizzes from teaching materials and evaluate those.

V. SUMMARY AND CONCLUSION

In Conclusion, Using automatic grading to assess student programs can increase students' motivation to practice programming continuously. It is important because programming requires continuous training.

There is a strong correlation between the frequency of doing exercises on programming assignments with the results of evaluations, both daily (quiz) evaluations, and middle and end of semester evaluations. Students who are "lack practice" is that very problematic in competency and scores at the middle and end of semester evaluation.

An integration between Online Judge as an automatic grading system and Moodle as an LMS allows teachers to know trends of students' competency immediately. It allows them to take actions immediately in improving teaching strategy. Students who get difficulties or other problems, can be easier to know and give feedback or assistance, or certain interventions.

More than 90% of students see the importance of the automatic grading system being involved in the teaching process, because students can practice anytime and anywhere. Motivation to practice gained by succeeding in solving problems and they know the score immediately. It leads students to work on the next questions. For some students, difficult questions are an interesting challenge to work on. This is another reason, that by giving training or assignments using an automatic grading, students realize that they must continue to learn and practice.

REFERENCES

- [1] Truog N, Roe P and Peter Bancroft P 2004 Static Analysis of Students' Java Programs. Proc. The Sixth Australian Computing Education Conference (ACE2004)
- [2] Liem I 2004 Aspek Pedagogi Pengajaran Pemrograman Pertama (online book Depdiknas-Republik Indonesia)
- [3] Yagci M 2016 Blended Learning Experience in a Programming Language Course and the Effect of the Thinking Styles of the Students on Success and Motivation (The Turkish Online Journal of Education Technology) volume 15 issue 4
- [4] Kinnunen P 2009 Challenges of Teaching and Studying Programming at a University of Technology – Viewpoints of Student, Teacher and the University (TKK Research Reports in Computer Science and Engineering A TKK-CSE-A4/09 Espoo)

- [5] Kitaya H, Inoue U 2016 An Online Automated Scoring System for Java Programming Assignments (Int. Journ. of Information and Education Technology) Vol. 6, No. 4
- [6] Wasik S, Antczak M, Badura J, Laskowski A, Sternal T 2016 A Survey on Online Judge System and their Applications (ACM Computing Surveys), Vol 1, No.1
- [7] Hernan L I, Pareja FC, Velazquez-I J.A 2010 Pedagogical Use of Automatic Grading (Edited by Mary Beth Rosson. Intech)
- [8] Fernando J, Liem I 2014 Components and Architectural Design of an Autograder System Family (Olympiads in Informatics) Vol 8,69-79, Vilnius University, IOI.
- [9] Patil A 2010 Automatic Grading of Programming Assignments (Master's Projects). Paper 51. San Jose State University.
- [10] Kavitha R K, Jaisingh W A 2018 Study on the Student Experiences in Blended Learning Environments Int.Journ. of Recent Technology and Engineering (IJRTE), volume 7 Issue-48
- [11] Carman J M 2005 Blended Learning Design: Five Key Ingredients (Utah State University)
- [12] Mozeliuss P, Rydell C 2017 Problem Affecting Successful Implementation of Blended Learning in Higher Education – the Teacher Perspective ICTE Journ. 6(1):4-13 DOI:10.1515/ijcte-2017-0001
- [13] Wangenheim CGV, Hauck J CR, Demetrio M F, Pelle R, Cruzalves, Nda, CodeMaster, Barbosa H Azevedo LF 2018 Automatic Assessment and Grading of App Inventor and Snap! Programs (Informatic in Education, Vilnius university) Vol.17, No. 1, 117-150