

Exploration on the Impact of Online Supported Methods for Novice Programmers

Tze Ying Sim

Center for American Education

Sunway University

Bandar Sunway, Malaysia

Email: tzeyings@sunway.edu.my

Abstract—Programming is a challenging subject to students who are exploring it for the very first time. Common problems faced by the students are difficult to break down the main problem to sub-problems, providing a solution to the problem using specific steps and placing the different steps in the correct order to solve the problem. Three online supported methods are implemented in a programming 1 class across three semesters. They are online group discussion using Facebook group, the hour of code activities provided by code.org, and online journaling using Google Form. The impact of the three different are being discussed. It is observed that all the three methods increased students engagement towards the subject. Students posed questions, gave feedback and participated actively in the class. More study is needed to determine the impact of these methods toward students grade.

Keywords—mobile learning, technology, engineering education

1. Introduction

Programming proves to be a challenging subject to take. The core of programming can be divided into two parts, namely the suitable steps and the correct order. These two simple concepts prove to be a challenge to students who have yet to solve a problem in such details. One of the in-class activities was to have a student role play a blind 'robot'. The robot was blindfolded, spun, and followed through a series of instruction from the peers to get from location A to location B. This simple activity that would normally take less than a minute for the student to move between the two locations, on a day to day basis, took 45 minutes to complete. The students are not used to breaking the problems, into steps, putting these steps into the different order, and repeating the steps. Having overcome this hurdle, the students need to put their steps in a language that they are not familiar with.

This research takes place in the Programming 1 class. It is a four credit hour subject offered by the American Degree Transfer Program (ADTP), Sunway University, Malaysia. This program does not award the final degree. ADTP students will continue their studies in North America. The pre-requisite for Programming 1 is pre-calculus (or College Algebra). One semester consist of 14 weeks of class, 1-week mid-term break and 2 weeks of the final exam. The class size is 30 students on the average. From

the record for the past 6 semesters, the failure rate of the class is between 30% and 33%. The students taking this subject are mainly Engineering and Computer Science students. However, students from other majors are also allowed to take this paper. This number proves to be small as taking this subject is seen as a risk for their Cumulative Grade Percentage Average (CGPA), and thus impeding the transfer opportunities of the students.

Through the past three semesters, three different online tools were used to enhance the teaching and learning experience of the class. They are the completion of 20 hours of code, discussion using Facebook group, and writing an online journal entry after every class. The online journal entry is known as *My Learning Journal* (MLJ). The structure of this paper is as follow: section 2 presents the related work for this research, section 3 describes the work was done and the data source for analysis, section 4 details the attempts of different methods in different semesters, section 5 analyses the results of each implementation in helping the students. Finally, section 6 conclusion on the effectiveness of the various methods in influencing students to understand the various programming concepts would be provided.

2. Literature Review

In order to increase the communication between faculty and students, various methods had been explored, ranging from short message system (SMS) via cellular telephone, using forum in learn management system (LMS) and messaging through social network system (SNS) [1] [2]. Examples of SNS include Facebook, Twitter, MySpace, Google+ etc. The most popular SNS is Facebook with the participation of 80% of students in the institute of higher education in Malaysia. Studies have shown that students activities on Facebook correlate negatively to the academic grades [3] [4] [1]. However, from the communication perspective, Facebook is a more effective tool as compared to the learn management system [5] [6] [7].

Different research activities to enhance the learning of programming for novice programmers had been conducted [8] [9] [6] [10]. Researcher in Computer Science finds it important to find the suitable way methods in developing future Computer Scientist [11] [12]. It is also important to implement the suitable methods base on the class size, the resources available, and the technical ability of the course conveyer [13].

The importance of computer education has been highlighted by the core members of code.org. The core members of code.org advocates for computer education to be included in the Kindergarten to year-12 (K12) syllabus and have the subject to be included towards graduation [14]. This provides the basic programming knowledge to high school students. However there are also contradicting view to this idea [15] [16]. The hour of code activities, through the class activities and graphic programming, encourages K-12 students to code [17]. Other research to introduce graphic programming to students includes block programming using Scratch [18], and App Inventor [19]. This has yet to be noticed in the current Malaysia primary or secondary education system [20]. Most of the computer science related activities are still within the Computer Science club initiative. This is a disadvantage to ADTP students when they transfer to the universities in North America. Therefore, it is important that the ADTP students get a strong foundation during their study period in the program.

Journaling is a process where the writer gets reflect on certain content, make an assessment, discover challenges or opportunities [21] [22]. The earliest form of collecting written feedback from students are the muddiest card, where students write down the content that they did not understand [13] [23]. Socratives exit ticket [24] is a quiz to be given to students at the end of the class. It presents 3 questions to the students, firstly how well did they followed the class, the content covered and answering a teacher's question. Active learning by echo 360 [25] allows questions to be integrated into the teaching slides, and a question may be conducted at any time. However, these methods are either manual or require a subscription. Physical muddiest card and Socratic requires manual compilation to have either feedback for each student or class combined. Google Form or Excel Survey may be another option [26], as the same form can be used to avoid manual compilation. Echo360, on the other hand, requires a subscription.

3. Methodology

This section would first introduce the way the class is being conducted, followed by a description of the three different methods implemented in this research.

In terms of class design, the class would start with a short revision of the previous session, followed by the content to be covered for the day, and a review session at the end of the class. The coverage of content was divided into 4 to 6 units for each lesson. The class duration was 2 hours, and each unit is between 15 and 20 minutes long. Each unit consist of an explanation of a basic concept, discussion with a partner on how to solve a problem using the concept, and implementing it using Eclipse. A 10 minutes break was given after 55 minutes of class. The revision session was conducted by asking students question using a tissue pack. A question would be asked, and the tissue pack would be thrown to a student. The student with the tissue pack would need to answer the question, and then throw it to the next student. At times, the students are requested to answer questions, or write down what they learn and the problems that they are facing in the class.

TABLE 1. SEMESTER WHERE DIFFERENT METHODS WERE IMPLEMENTED

	Fall 2013	Spring 2014	Fall 2014	Spring 2015
Facebook Group	-	X	X	X
Hour of Code	-	X*	X	X
MLJ	-	-	-	X

X* end of the semester submission

As mentioned in section 2, a few ways of increase the communication between faculty and students had been looked into. As the faculty would like to keep their cellular numbers private, the option to use SMS and Whatsapp messaging system was excluded. The usage of the forum provided by the LMS during Fall 2013 did not attract the participation of students. These reasons narrow the selection down to the usage of SNS. Among the SNS looked into, Facebook is selected for its popularity. As the faculty were interested in keeping their personal account private, a special account was set up for this purpose. A survey was conducted at the end of the semester to access the impact of using Facebook as a communication tool in the class. Data from a survey and the grades from the students for the specific semesters were used to access the implementation of Facebook group.

The Hour of Code activities as introduced in section 2 was meant to expose the K12 students system to Computer Science. The presentation of activities in Hour of Code was game-based and video-based. Completing each level of the game allows the student to proceed to the next level. The 20 hour of code activity is introduced to help students understand the control structure of programming. This includes select statements and loop structure. The students are requested to work on the 20 hours of code activities on their own, outside of class time. Data from survey and class observation by the lecturer will be used to access the influence of the hour of code activity on the students.

As described in section 2, there are different ways to conduct written feedback. Journaling is a compilation of feedback by the students, collected after the classes conducted throughout the semester. The lecturers would read the entries, and provide additional explanation via the class Facebook group or during the next class. During week 12 of the semester, the lecturers printed the journal entries by students and provided individual feedback for the students. The data analysis for this research is based on both quantitative and qualitative survey. The sources for quantitative analysis were the participation list of students in the different activities, the test result, and the semester end survey. The qualitative survey included feedback from both the students and the lecturer. It would be ideal to compare the different cohorts general performance, as a control measure. However, this is not possible to date, as the data is not available.

4. Experiment

The three different methods presented in section 3 were implemented in different semesters. Table 1 lists the different methods implemented through the different semester.

At the beginning of each semester, the lecturer will create a Facebook group for the class. The group would be established as an open group. The lecturer would need to add a few students who are in his/her friends' list and the students are able to add their friends into the group. Apart from that, the name of the group would be made known during the class, and the students can find the group and add themselves to the group. This activity was conducted during the first class in the semester. At the end of the second week, the status of the group was then modified to close group. Students and lecturers posted and replied as and when needed. Official announcements were done both in the LMS as well as the Facebook group. No reward was linked to the participation of Facebook group discussion. The survey for Fall Semester 2014 was modified to capture more information as compared to the survey for Spring Semester 2014.

The students were introduced to the Hour of Code activities during the first class. They were given the code to enroll to the class. The lecturer would be able to monitor the progress of the students. During the first implementation in Spring Semester 2014, the students were given until the end of the semester to complete the tasks. The students would complete the activity around week 13 and week 14. During the Fall Semester 2014, and Summer Semester 2015, students were required to complete the activity during the first two weeks of class. The selection statement was introduced during the third week of class and the control structures during the 4th week of class. Students who completed the 20 Hours of Code activity will be awarded 2.5 marks extra credit.

During Spring Semester 2014, and Fall Semester 2014 feedback from students were collected through Socrative exit ticket. The students would answer the 3 exit ticket questions at the end of every class. However, it was tedious to compile data for the whole semester. During Spring 2015, MLJ was implemented. Instead of answering Socrative exit ticket, students were requested to write the content that they deemed as important, the challenging content for the day, and reflection for the class. The entries were tracked by student identification number and name. During week 12, the entries by the students were printed individually and returned in the class. The lecturer provided individual feedback to the students. There were no rewards nor penalties involved in the implementation of MLJ.

5. Results and Discussion

The results for the three methods discussed in section 4 will be presented in the following discussion.

5.1. Facebook Group

The implementation of Facebook group had positively influenced the engagement and the participation of the students in the class and outside the class. The survey conducted in Spring Semester 2014 had only two questions concerning the implementation of Facebook Group. The result of the survey is shown in Table 2. The whole class agreed that Facebook group encourages the class discussion and students working on the course content. Three more questions were added to the survey in Fall

TABLE 2. STUDENTS' FEEDBACK ON FACEBOOK GROUP TECHNIQUE IN SPRING 2014

Spring 2014	Strongly Agree	Agree	Neutral	Dis-agree	Strongly Dis-agree
Class Discussion	19	12	N/A	N/A	N/A
Work On Course Content	14	17	N/A	N/A	N/A

TABLE 3. STUDENTS' FEEDBACK ON FACEBOOK GROUP TECHNIQUE IN FALL 2014

Fall 2014	Strongly Agree	Agree	Neutral	Dis-agree	Strongly Dis-agree
Class Discussion	13	8	0	1	0
Raise Question	12	7	2	1	0
Interactive Class	12	7	2	1	0
Work On Course Content	11	5	5	1	0
Out Of Class Engagement	12	6	2	2	0

Semester 2014 (see Table 3). From the survey conducted it was observed that 85% of the class agreed that having a Facebook Discussion group encourages the class discussion, encourages them to raise question in the class, encourages the class to be more interactive, encourages them to work on the course content and encourages them to be engaged with the class outside class hours.

It was noticed that students post problems that they have in the class, and other students answering to their queries. Apart from that students would also post interesting videos and documents that were relevant to the class content to be shared with other classmates. However, when analyzing the activity in Facebook group with their grades, there was no significant correlation found.

5.2. Hour of Code

As mentioned in the introductory paragraph, the failure rate for the class was 33%. During the Spring Semester 2014, the students were introduced to Hour of Code. However, they were given until the end of the semester to complete this task. It was observed that about 17% (6 out of 31) students were still having problems with the blocks. Common mistakes include having a method within a method and having a nested block outside the method. The students were also having problems with the control structures, especially with defining nested loops, and putting statements that should be in a loop outside the loop.

During the two semesters, where the students were required to complete the Hour of Code task within the first two weeks of class. It was observed that fewer students, namely 10% (3 out of 30) students made the mistakes committed earlier. This showed that by completing the Hour of Code activity, students were able to understand the block concept as well as the control structure better. A survey was conducted during the end of Spring Semester 2015 and the result of the survey was shown in Table 4. It was observed that 87% of the students agreed or

TABLE 4. STUDENTS' FEEDBACK ON THE USAGE OF HOUR OF CODE FOR FALL 2014

Doing the hour of Code helps me to:	Strongly Agree	Agree	Neutral	Dis-agree	Strongly Dis-agree
understand the control structure better	7	13	2	0	1
understand the blocks concepts better	9	13	1	0	0
to be more interested in programming	6	11	5	1	0

strongly agreed that hour of code helped them understand the control structure better, and 96% agreed that it helped them understand the block concepts better.

When comparing to the Test 1 result for Spring Semester 2015, 21 out of 30 students scored more than 7.4 out of the 15 marks. The average mark for Test 1 was 8.5. When taking the average 33% failure rate as the bench mark, no improvement was observed. However, when evaluating the content of the test, students did not lose mark by committing mistakes related to blocks. The impact of Hour of Code was mainly observed in the practical questions.

5.3. Mobile Learning Journal

MLJ was implemented for only one semester, namely Spring Semester 2015. Students were requested to key in their reflection after every class. According to the participation of MLJ entries record, it was observed that the participation at the beginning of the semester was slightly higher as compared to the end of the semester (see Figure 5.3). It should be highlighted that students were not requested to do MLJ entries on the following dates 20/5, 25/5, 1/6, 24/6 and 1/7. However, there were still entries for these dates, as the students took the initiatives to write the entries on their own. The withdrawal week was in the first week of June. About 7 students withdrew from the subject, explaining the decrease in participation. Generally, it was observed that the participation of MLJ was encouraging with more than 87% of the students participating consistently. When asked in the survey on the frequency of MLJ, 78% mentioned that it should be after every class, and another 22% agreed that it should be after every week. This was summarized in Table 5.

As shown in Table 6, 87% of the students strongly agreed or agreed that the MLJ helps them to reflect on the content covered in the class. 87% of the students also either strongly agreed or agreed that receiving feedback on MLJ motivates them to work on the course content. This help to draw the initial conclusion that MLJ is a good tool to help the student to reflect on the course content and highlight the challenging content covered.

The lecturer provided feedback that MLJ helped to provide instant feedback for the class conducted. The content covered helped to summarize content that still stays with the students at the end of the class. This helps to gauge the learning outcomes observed by the students. The challenging content would be highlighted during the

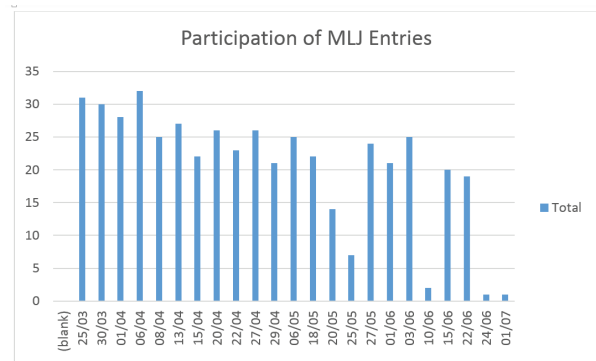


Figure 1. Participation of MLJ Entries.

TABLE 5. STUDENTS' FEEDBACK ON THE FREQUENCY TO JOURNAL

	every class	every week	every fortnight	after the test
The gap for MLJ should be after	18	5	0	0

TABLE 6. STUDENTS' FEEDBACK ON THE USAGE OF MLJ

	Strongly Agree	Agree	Neutral	Dis-agree	Strongly Dis-agree
The usage of MLJ helps me to reflect on the content covered in the class	10	10	3	0	0
Receiving Feedback on MLJ motivates me to work on the course content	7	13	3	0	0

revision session at the beginning of next class. From the class reflection section, various feedback and suggestion can be observed. Among suggestion adopted from MLJ was to consistently have the 10 minutes break for the students to freshen up, get a drink or buy some food. The feedback provides an insight on what is happening in the students life. An example of feedback is "Miss thank you for not penalizing me for dozing off in your class. I was sick through the night." MLJ helped the lecturer to understand the students better, and thus would be able to tailor the class according to its needs.

6. Conclusion

This research reflected on three online tools implemented to support the teaching and learning of programming 1. The strength of each tool is as follows:

- Facebook group is a good communication tool for students to raise question and have discussion
- 20 Hours of Code activity is suitable to help students understand the concept of programming blocks, and control structure
- MLJ helps students to reflect on the content covered, and help the lecturer to revise content that posed to be a challenge to the students. This

can also work as a feedback for every class session conducted.

The implementation of all three tools managed to increase the engagement of teaching and learning in the class. However, the impact on tests grades are not significant. This could be due to the contrast of activities between in-class, and the test. In-class activities focus more on solving-problem and hands-on programming, through discussion and group programming. Whereas the students are expected to solve the test on their own, and they tend to panic when the program have syntax error during the test. A more detailed study on the relevance and details of both theory and practical evaluation can help us access our students better.

Acknowledgments

The authors would like to thank the students from programming 1 who responded to the survey.

References

- [1] R. Junco, "The relationship between frequency of facebook use, participation in facebook activities, and student engagement," *Computers and Education*, vol. 58, pp. 162–171, 2012.
- [2] T. Y. Sim, S. L. Lau, P. Zipf, and K. Kimm, "Design and development of a supported tiered software for teaching and learning using a connected mobile learning application," *World Applied Science Journal*, vol. 30, no. 30, pp. 247–255, 2014.
- [3] W. W. Goh, "Students' behavior and perception of using facebook as a learning tool," *2013 8th International Conference on Computer Science & Education*, no. Iccse, pp. 731–736, 4 2013.
- [4] S. Jamil, F. Zehra, R. Naqvi, and S. Bhamani, "Impact of facebook intensity on academic grades of private university students," in *5th International Conference on Information and Communication Technologies p. 1-10*, 2013, pp. 1–10.
- [5] M. Maleko, D. Nandi, M. Hamilton, D. D'Souza, and J. Harland, "Facebook versus blackboard for supporting the learning of programming in a fully online course: The changing face of computing education," *2013 Learning and Teaching in Computing and Engineering*, pp. 83–89, 3 2013.
- [6] T. Y. Sim and B. Vogel-Heuser, "Reviews and findings on implementing active learning in a large class environment for mechatronics and computer science students," *Education Engineering (EDUCON), 2010 IEEE*, 2010.
- [7] M. Maleko, M. Hamilton, and D. D'Souza, "Access to mobile learning for novice programmers via social networking sites," *2012 7th International Conference on Computer Science & Education (ICCSE)*, no. Iccse, pp. 1533–1538, 7 2012.
- [8] I. Boada, J. Soler, F. Prados, and J. Poch, "A teaching/learning support tool for introductory programming courses," *Information Technology Based Proceedings of the Fifth International Conference on Higher Education and Training, 2004. ITHET 2004.*, pp. 604–609, 2004.
- [9] O. Hazzan and Y. Dubinsky, "Teaching a software development methodology: the case of extreme programming," *Proceedings 16th Conference on Software Engineering Education and Training, 2003. (CSEE&T 2003).*, 2003.
- [10] H. Zhu and M. Zhou, "Methodology first and language second: A way to teach object-oriented programming," in *Companion of the 18th Annual ACM SIGPLAN Conference on Object-oriented Programming, Systems, Languages, and Applications*, ser. OOPSLA '03. New York, NY, USA: ACM, 2003, pp. 140–147. [Online]. Available: <http://doi.acm.org/10.1145/949344.949389>
- [11] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, "A study of the difficulties of novice programmers," *ACM SIGCSE Bulletin*, vol. 37, no. 3, pp. 14–18, 2005.
- [12] A. Pears, S. Seidman, L. Malmi, L. Mannila, E. Adams, J. Benned- sen, M. Devlin, and J. Paterson, "A survey of literature on the teaching of introductory programming," *SIGCSE Bulletin*, vol. 39, no. 4, pp. 204–223, 2007.
- [13] T. Sim, *Effectiveness of the Methods for Engineering Courses in a Large Non-homogenous Class Setting: With Regards to the Specific Disciplines - Computer Science and Mechatronics in Learning Embedded System*, ser. Embedded System II Forschung. Kassel University Press, 2010. [Online]. Available: https://books.google.co.uk/books?id=QC_WzN9y8iAC
- [14] H. Partovi, A. Partovi, and C. Vilson, "Anybody can learn — code.org," <http://code.org/>, 2015, [Online; accessed 12-August-2015].
- [15] S. Fincher, "What are we doing when we teach programming?" *FIE'99 Frontiers in Education. 29th Annual Frontiers in Education Conference. Designing the Future of Science and Engineering Education. Conference Proceedings (IEEE Cat. No.99CH37011*, vol. 1, 1999.
- [16] M. Guzdial, "Why the u.s. is not ready for mandatory cs education," *Communications of the ACM*, vol. 57, no. 8, pp. 8–9, 2014.
- [17] T. Ball and B. Zorn, "Teach foundational language principles," *Communications of the ACM*, vol. 58, no. 5, pp. 30–31, 2015.
- [18] N. Available, "Scratch - imagine, program, share," <http://code.org/>, 2015, [Online; accessed 12-August-2015].
- [19] MIT, "Mit app inventor — explore mit app inventor," <http://scratch.mit.edu/>, 2015, [Online; accessed 12-August-2015].
- [20] M. Malaysia Education Blueprint, "Malaysia education blueprint 2013 - 2025," Tech. Rep., 2013.
- [21] P. J. Denning, "The social life of innovation," *Communications of the ACM*, vol. 47, no. 4, p. 15, 2004.
- [22] J. Robertson, "Rethinking how to teach programming to newcomers," *Communications of the ACM*, vol. 57, no. 5, pp. 18–19, 2014.
- [23] V. Simpson-Beck, "Assessing classroom assessment techniques," *Active Learning in Higher Education*, vol. 12, no. 2, pp. 125–132, 2011.
- [24] Socrative, "Socrative," <http://socrative.com/>, 2015, [Online; accessed 12-August-2015].
- [25] Echo360, "Higher educations first active learning platform — echo360," <http://echo360.com/>, 2015, [Online; accessed 12-August-2015].
- [26] T. Y. Sim, "My learning journal," <http://bit.ly/1TsbydJ/>, 2015, [Online; accessed 12-August-2015].