YONG, S.T., TIONG, K.M., CHAN, A. and KHIEW, P.S. 2021. A flipped classroom: learning experiences in programming. *International journal of virtual and personal learning environments* [online], 11(1), pages 23-37. Available from: https://doi.org/10.4018/IJVPLE.2021010102

A flipped classroom: learning experiences in programming.

YONG, S.T., TIONG, K.M., CHAN, A. and KHIEW, P.S.

2021

International Journal of Virtual and Personal Learning Environments

Volume 11 • Issue 1 • January-June 2021 • ISSN: 1947-8518 • eISSN: 1947-8526

IGI PUBLISHING
AN IMPRINT OF IGI GLOBAL
WWW.IGI-GLOBAL.COM

Call for Articles

International Journal of Virtual and Personal Learning Environments

Volume 11 • Issue 1 • January-June 2021 • ISSN: 1947-8518 • eISSN: 1947-8526

MISSION

The mission of the **International Journal of Virtual and Personal Learning Environments (IJVPLE)** is to study and disseminate research about the design, development, and evaluation of online learning environments. IJVPLE is committed to encouraging the best teaching and learning practices by examining the role of technology enhanced learning in the emerging area of virtual and personal learning environments.

COVERAGE AND MAJOR TOPICS

The topics of interest in this journal include, but are not limited to:

3D online worlds • Adaptive and intuitive environments • Applications of the semantic web • Blended Learning • Collaborative Learning • Computer aided language learning (CALL) • Computer mediated communication • Computer mediated conferencing • E-Assessment • E-Learning • E-portfolios • Educational mashups • Emerging Technologies • Gaming and learning • Intelligent content • Knowledge management and learning • Lifelong Learning • Mobile learning environments and applications • Multimedia applications and virtual reality • Networks/grids for learning • Platforms and authoring tools • Remote and virtual laboratories • Responsive environments • Virtual Learning Environments • Web-based learning (WBL)

ALL INQUIRIES REGARDING IJVPLE SHOULD BE DIRECTED TO THE ATTENTION OF:

Thomas Connolly, Editor-in-Chief • IJVPLE@igi-global.com

ALL MANUSCRIPT SUBMISSIONS TO IJVPLE SHOULD BE SENT THROUGH THE ONLINE SUBMISSION SYSTEM:

http://www.igi-global.com/authorseditors/titlesubmission/newproject.aspx

IDEAS FOR SPECIAL THEME ISSUES MAY BE SUBMITTED TO THE EDITOR(S)-IN-CHIEF

PLEASE RECOMMEND THIS PUBLICATION TO YOUR LIBRARIAN

For a convenient easy-to-use library recommendation form, please visit:
http://www.igi-global.com/IJVPLE

Table of Contents

International Journal of Virtual and Personal Learning Environments

Volume 11 • Issue 1 • January-June-2021 • ISSN: 1947-8518 • eISSN: 1947-8526

Research Articles

The *International Journal of Virtual and Personal Learning Environments* is indexed or listed in the following: ACM Digital Library; Bacon's Media Directory; Cabell's Directories; DBLP; ERIC – Education Resources Information Center; Google Scholar; INSPEC; JournalTOCs; Library & Information Science Abstracts (LISA); MediaFinder; SCOPUS; The Standard Periodical Directory; UGC-CARE List (India); Ulrich's Periodicals Directory

International Journal of Virtual and Personal Learning Environments
Volume 11 • Issue 1 • January-June 2021

A Flipped Classroom:
Learning Experiences in Programming

Su Ting Yong, University of Nottingham Malaysia, Malaysia
Kung Ming Tiong, University of Nottingham Malaysia, Malaysia
Andy Chan, University of Nottingham Malaysia, Malaysia
https://orcid.org/0000-0003-2267-4949
Poi Sim Khiew, University of Nottingham Malaysia, Malaysia

ABSTRACT

This study explored students' perceptions of a flipped classroom for an introductory programming class. Students were required to watch video lectures and read lecture notes in advance (pre-class self-study) to prepare themselves for the in-class lectures and tutorials. A mix-methods approach was employed: quantitative survey (n=204) and qualitative interview (n=7) were administered simultaneously. The results suggested that students are not fully ready for a flipped classroom. Most of the students still prefer face-to-face in-class lectures and tutorials. The in-class activities have a positive impact on students' test performance, especially the male students. Peer learning however induces a negative impact on students' test performance, especially among the female students. Pre-class self-study has no impact on students' test performance, except for those without prior programming experience. Females outperform males even though they lack prior programming experience. Students, regardless of programming background, respond equally to a flipped classroom approach.

KEYWORDS

Flipped Classroom, In-Class Lecture, In-Class Time, In-Class Tutorial, Learning Approaches, Pre-Class Preparation, Programming, Video Lectures

INTRODUCTION

The flipped classroom is a pedagogical model in which the usual lecture and follow-up learning activities are reversed. In a flipped classroom, the lecture materials are delivered online (e.g. video lectures and lecture notes) prior to the face-to-face class time. Then during the in-class time, students will involve in the active learning tasks, e.g. problem-solving, hands-on exercises and group discussions. Flipped classrooms embody a set of learning theories such as active learning, peer-assisted learning and collaborative learning (Akçayır & Akçayır, 2018). In this study, the authors have reported the evaluation of a flipped programming class. Computer programming is one of the fundamental skills in Computer Science (CS). The flipped classroom model is hoped to improve students' learning experience replacing the conventional approach to programming pedagogy that was reported to be boring and monotonous (Jenkins, 2002; Maragos & Grigoriadou, 2007).

DOI: 10.4018/IJVPLE.2021010102

The authors focused on students' learning experience, particularly the effectiveness of flipped learning. The flipped classroom was designed after Awidi and Paynter (2019), in which the flipped sessions comprised pre-recorded lectures and online quizzes, while retaining the in-class lectures and activities. The authors assumed that different flipped sessions and flipped materials (independent variables) would be the key drivers for students' test performance (dependent variable). This study employed two designs embraced by *ex post facto* research: (1) the correlational model; and (2) the criterion group model. The primary research methodology used in this study was quantitative in nature (e.g. survey and programming test). Then, qualitative interview was used to complement the quantitative results. This paper started with a review of relevant literature, followed by the description of the aims and objectives, methodology, results and discussions, and conclusion.

RELATED WORK

The Flipped Classroom

The flipped classroom is a new pedagogical model, in which the typical in-class activities (e.g. lecture) and out-of-class activities (e.g. homework) are reversed (Akçayır & Akçayır, 2018; Awidi & Paynter, 2019; Lacher & Lewis, 2015). A flipped classroom is a way to extend learning beyond the classroom, and the in-class time is reserved for meaningful and practical learning activities such as interactive discussions, group learning activities, hands-on exercises and problem-solving activities (Giannakos, Krogstie, & Chrisochoides, 2014; Lacher & Lewis, 2015). The content knowledge is given out-of-class with electronic resources such as video lectures, online materials and interactive quizzes (Akçayır & Akçayır, 2018). A meta-analysis study has reviewed 71 studies related to flipped classrooms and identified four major advantages of flipped learning - improved students' performance, positive students' feedback, improved students' satisfaction and enhanced learning flexibility (Akçayır & Akçayır, 2018). Some major drawbacks include time-consuming (e.g. video recording or editing), poor quality of videos and students' limited pre-class preparation time (Akçayır & Akçayır, 2018). The researchers have further explained that it is an extra burden for the students to watch video lectures before every class because they have to put in more efforts and time (Akçayır & Akçayır, 2018). Another controversial argument pointing towards the flexibility of video lectures has reduced the face-to-face classroom interaction between the instructors and students, it would be more to the disadvantage of weak students (Ng, 2018).

The Flipped Classroom in Computer Science and Programming

The flipped classroom approach has been used in many different disciplines, including CS (Dazo, Stepanek, Fulkerson, & Dorn, 2016; Giannakos et al., 2014; Lacher & Lewis, 2015), computer programming (Baldwin, 2015; Isomöttönen & Tirronen, 2016; Kim & Kim, 2017; Lacher & Lewis, 2015), biology (Awidi & Paynter, 2019), nursing (Barbour & Schuessler, 2019; Njie-Carr et al., 2017; Tan, Yue, & Fu, 2017), and mathematics (Lopes & Soares, 2018). According to Giannakos et al. (2014), more than 30 studies on flipped classrooms in CS have positively impacted on students' performance, attitudes and engagement. Students' performance in a flipped classroom is equivalent or even better than the conventional approach (Giannakos et al., 2014). For instance, in Lacher and Lewis (2015), a flipped classroom has helped the surface learners to improve their programming grades. According to Giannakos et al. (2014), flipped learning engages CS students in active learning and activates higher-order thinking. Flipped learning also promotes computational thinking in programming education (Kim & Kim, 2017). In a flipped classroom model, students can spend more in-class time on hands-on activities (e.g. programming questions) and consult the instructor if they need any help (Lacher & Lewis, 2015). Despite the many advantages of flipped learning, the success of a flipped classroom depends heavily on students' pre-class preparation (Dazo et al., 2016; Lacher & Lewis, 2015).

Although CS instructors are expected to be more competent and confident in preparing video lectures, they do face difficulties in flipped classrooms. According to Dazo et al. (2016) and Giannakos et al. (2014), CS instructors have to spend a huge amount of time recording video lectures, but many students do not watch the videos (whether before or after the class). On the contrary, students who watch the videos tend to be absent from the class more frequently (Giannakos et al., 2014). The outcome of the flipped learning is very much depending on the nature of the subject taught. In flipped programming classrooms, most of the students claim to benefit more from the in-class activities such as practical sessions, hands-on problem-solving and laboratory exercises (Baldwin, 2015; Piteira & Costa, 2013). Although flipped materials such as sample programs, online tutorials and video lectures are useful (Piteira & Costa, 2013), many students find it hard to learn programming independently without going to class (Baldwin, 2015).

Gender and Programming Background Differences

As with any pedagogical strategy, the outcome of a flipped classroom is affected by the nature of the subject taught and students' background, e.g. gender and prior programming experience. In most of the countries surveyed, CS is a discipline dominated by men (Köppe & Bartilla, 2017). In another study conducted at United Kingdom, Wagner (2016) has found that females are awarded significantly fewer first-class degrees than males. Furthermore, females have significantly less pre-college programming experience than their male counterparts (Murphy et al., 2006). Studies have shown that there are significant gender differences in programming self-efficacy, confidence, enthusiasm, usage of features and technical problem-solving abilities (Burnett et al., 2010; Marsh, 2010). Despite the differences, this does not suggest that females are less proficient in programming (Burnett et al., 2010). There is no correlation between students' prior programming experience and their grades in the university (Murphy et al., 2006). Furthermore, females can write computer programs as good as males (Murphy et al., 2006). Although gender does not show any significant effect on flipped learning readiness (Hao, 2016), the above individual differences deserve investigation.

AIMS AND OBJECTIVES

This study aimed to explore students' learning experience in a flipped programming classroom. The study assessed both perceptions (how the students felt about flipped learning?) and performance (how well the students could program?). The research questions (RQs) of this study are:

RQ1: How students perceive the effectiveness of various flipped sessions?
RQ2: What is the relationship between various flipped sessions and students' test performance?
RQ3: What are the flipped materials used to learn programming?
RQ4: What is the relationship between various flipped materials and students' test performance?
RQ5: Do gender differences affect students' test performance?
RQ6: Do programming background differences affect students' test performance?

METHODOLOGY

Research Method

There are a few quantitative methodologies that can be used in educational research such as experimental, *ex post facto*, survey, and correlational (Johnson & Christensen, 2008). This study employed the correlational and criterion group models embraced by *ex post facto* research. Experimental research was not appropriate due to ethical considerations. It was ethically undesirable to isolate one group of students (control group) and treated them differently from others, e.g. attended conventional classrooms and had no access to flipped materials. Being educators, it was ethically wrong to control or manipulate the dependent variable (test performance) by causing the control group to become failures or dropouts. In this case, *ex post facto* research was particularly more appropriate. *Ex post facto* research is an exploratory tool that yields useful information concerning the nature of phenomena (Cohen, Manion, & Morrison, 2007). This approach is appropriate in educational contexts where the independent variables (e.g. gender and programming background) lie outside of the researchers' control (Cohen et al., 2007).

In this study, the authors took the effect or dependent variable (test performance) and examined the data retrospectively to establish causes and independent variables. Two approaches embraced by *ex post facto* research were used:

1. The correlational model – identified the association of a flipped classroom (independent variable) and test performance (dependent variable).
2. The criterion group model – examined the test performance (independent variable) of two different groups and tried to account for the differences by investigating possible antecedents. Two categorical data were used, i.e. gender and programming background.

The Flipped Programming Classroom

MATLAB was the first programming module taken by a group of foundation students (age 18) at a private university in Malaysia. The module aimed to provide students with the basic understanding of computer programming. In the module, students were expected to write computer programs using functions and script files to solve engineering and mathematical problems. The module consisted of a weekly two hours in-class lecture and a weekly one hour in-class tutorial in a laboratory setting. Students took the module over a period of 10 weeks. The module was taught using a flipped classroom approach. A full set of *lecture notes* and *video lectures*, done as screencasts were posted on Moodle. Students were required to read the lecture notes and watch the videos in advance (*pre-class self-study*), to prepare themselves for the in-class lectures and tutorials. Each video lecture was approximately 30-45 minutes long. The video lectures aimed to provide students with the basic and fundamental programming concepts. During the *in-class lecture*, the instructor explained and summarized the programming concepts. The in-class lecture was conducted in the form of a hands-on workshop. Students learned how to design, write and debug computer programs. They could examine sample programs, test alternative program solutions, and discuss or debate the new programming concepts. During the *in-class tutorial*, students attempted the hands-on *tutorial questions* available in Moodle. Students could learn together with their peers (*peer learning*) or ask the instructor for any problems and enquiries. In both the in-class lecture and tutorial sessions, the lecturer took on the role of facilitator to encourage students to engage in deep and active learning. After the in-class lecture and tutorial, students could revise and test their knowledge and understanding through *online quizzes* available in Moodle. The online quizzes comprised multiple-choice and short programming questions.

Data Collection

Two types of data were collected: test performance and students' perceptions data. Upon completion of the 10-week course, students were given a 2.5-hour test on MATLAB (dependent variable). The test consisted of 50 multiple choice questions (50%) and 25 short programming questions (50%). The test performance out of 100% was recorded. Then, surveys and interviews were administered to explore students' learning experience in a flipped classroom. A mix-methods approach was employed, in which quantitative survey and qualitative interview were administered simultaneously. The quantitative survey (5-point Likert scale from 1-strongly disagree to 5-strongly agree) was used to provide a general understanding of students' sample by investigating the responses to a flipped classroom (independent variables). The survey was administered online, and the response rate was approximately 89% in which 204 students responded. The survey took approximately 10-15 minutes to complete. Qualitative interviews were conducted to obtain the detailed understanding of individual students. The issues covered in both surveys and interviews were similar, but the conversational nature of the interviews allowed flexible responses from the students to elaborate on the issues. A sub-sample of students who participated in the survey was selected (n=7) for an interview based on voluntary basis. Each interview took approximately 30-45 minutes to complete and was administered face-to-face. All the interviews were voice recorded and transcribed into text data. Three major areas were explored in the surveys and interviews.

1. Background: gender and prior programming experience
2. Flipped sessions: The following sessions are useful in helping me to learn programming effectively - (a) pre-class self-study, (b) in-class lecture, (c) in-class tutorial and (d) peer learning.
3. Flipped materials: I have used the following materials to learn programming effectively - (a) video lectures, (b) lecture notes, (c) tutorial questions and (d) online quizzes.

Data Analyses

The quantitative data was analyzed using SPSS version 25. Descriptive statistics, correlation analysis and Mann-Whitney test were performed. Descriptive statistics was used to summarize and present the quantitative data in a tabular form, e.g. percentages of agreement or disagreement. Spearman correlation analysis was used to measure the strength and direction of a monotonic relationship between the test performance and other variables measured on ordinal scales (e.g. pre-class self-study, in-class lecture, hands-on tutorial exercises, peer learning, etc.). The analysis was performed to establish if there were possible connections between test performance and flipped sessions or materials. A strong correlation indicated that the variable had an important influence on the test performance. The Mann-Whitney U test was used to compare differences between two categorial groups (e.g. males or females, with or without prior programming experience) when the dependent variable was an ordinal data or a continuous data that was not distributed normally.

The qualitative data was analyzed manually using the content analysis approach. Two types of coding systems were used during the process of segmenting and coding, *a priori codes* and inductive codes. *A priori codes* were developed based on the research questions (e.g. flipped sessions and materials). Then, inductive codes were generated from the data. Finally, a composite description of the essence of the experience for all the students was developed, e.g. Student 1-7. In the results and discussions section, only the significant and selected verbatims were included.

RESULTS AND DISCUSSIONS

RQ1: How Students Perceive the Effectiveness of Various Flipped Sessions?

In Table 1, students rated the effectiveness of various flipped sessions. About 94% of the students claimed that in-class activities (lecture and tutorial) were the most effective approaches. During the interview, six out of seven students agreed on this. There were three main reasons why in-class

Table 1. Students' responses to the effectiveness of various flipped sessions

Flipped Sessions	Disagree %	Neutral %	Agree %
Pre-class self-study	19	25	56
In-class lecture	2	4	94
In-class tutorial	1	5	94
Peer learning	11	11	78

lecture was beneficial for the students. (1) The lecturer gave a thorough explanation. Student 1 said, "Lecture is useful. Because the lecturer will explain the step one-by-one. In the lecture note, there is some skips in the information. During the lecture, the lecturer will teach step-by-step and by changing the basic program to the advance one. So, it will make us easier to understand". (2) Students learned troubleshooting with the help of the lecturer. Student 2 said, "If I got any problems and the programs cannot run properly, I can ask the lecturer". (3) The lecturer answered all their queries. Student 3 said, "Lecture is important because you need to know why you want to do this? Why you want to do that?" From the interviews, it could be seen that students valued the presence of a lecturer, e.g. to guide them and to facilitate learning. In-class lectures could not be entirely replaced by video lectures. A lecturer was much more than giving lectures; a lecturer could mentor, motivate and inspire students.

There were two main reasons why in-class tutorial was beneficial for the students. (1) Hands-on computer training was important. Student 4 said, "Do more exercises is more effective way to study programs. I need to do exercise to understand…If you do not do enough exercise, you might forget". (2) Promoted creative thinking. Student 1 said, "The questions will make me think how to solve the problems because there are many ways to solve one question… So can learn different ways to solve one question". This finding concurs with Piteira and Costa (2013). Students learned programming by writing programs – planning, analyzing, designing, coding and debugging. They devised a programming strategy, reflected on the feedback (e.g. syntax or logic errors), regenerated a better strategy, and tested and re-evaluated the strategy again. Programming knowledge was constructed by solving hands-on programming questions.

Peer learning (78%) was rated to be effective too. Nevertheless, there were two sides of the fence. (1) Pros: students learned from each other. Student 3 said, "The concept that you feel like very weird or something, then this you can ask your friend and see how your friends think about it. So, you can understand from their perspectives and maybe you can learn from that". (2) Cons: students confused each other. Student 5 said, "My friends might have different type of thinking and different type of approach. They might confuse you… the way they ask the questions may also confuse you". From the interviews, it could be seen that peer learning should be used with caution, especially among the novice learners.

Pre-class self-study (56%) was rated as the least effective session. Students revealed that they would prefer post-class revision (self-study) rather than pre-class preparation (self-study). Student 2 said, "If the lecturers have answered all my doubts during the class, it is better for me to study alone". It seemed that students were not fully engaged with flipped learning (i.e. prepared for class).

RQ2: What is the Relationship Between Various Flipped Sessions and Students' Test Performance?

There were significant correlations between certain flipped sessions and test performance (see Table 2). In-class activities (lecture and tutorial) were weakly positive correlated with test performance (lecture: $r_s = .189$; tutorial: $r_s = .184$, $p < .01$). The finding indicated that students who learned programming through in-class activities were more likely to perform better in the test. On the contrary, peer learning was weakly negative correlated with test performance ($r_s = -.161$, $p < .05$). The finding implied that

Table 2. Correlations of various flipped sessions and test performance

	Pre-class self-Study	In-class lecture	In-class tutorial	Peer Learning
Correlation Coefficient	0.094	.189**	.184**	-.161*
Sig. (2-tailed)	.176	.006	.008	.020
*. Correlation is significant at the 0.05 level (2-tailed). **. Correlation is significant at the 0.01 level (2-tailed).				

students who learned together with peers tended to perform worse in the test. And finally, there was no significant correlation between pre-class self-study and test performance. This finding contradicted with Dazo et al. (2016). In this study, correlation analysis suggested that students' programming knowledge could be improved by attending lectures and tutorials. On the other hand, peer learning should be minimized to avoid possible adverse effects on test performance.

RQ3: What are the Flipped Materials Used to Learn Programming?

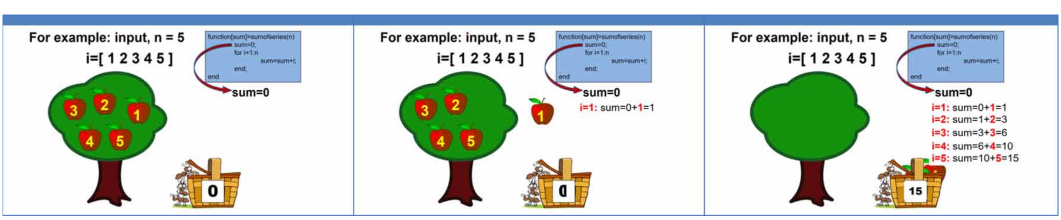
In Table 3, students rated the usefulness of flipped materials. About 99% of the students revealed that they used lecture notes to learn. Lecture notes were the main sources of reference. Student 1 said, "lecture note is the most useful for learning", Student 2 said "Lecture notes gives me the most information", Student 5 said, "Lecture note is like a dictionary. If you don't understand, just search the lecture notes" and Student 7 said, "I mostly understand through the lecture notes".

The sample programs (86%) and diagrams (87%) (in the lecture notes) were claimed to be useful too. (1) Sample programs helped students to learn by examples. Student 1 said, "The sample program is useful because when I typing the sample program, I will also understand the use of the functions", Student 3 said, "The sample program is like let me know how complex it could be… So I can use the concept to turn into my own or something like that" and Student 6 said, "Lecture notes should not have much theory but more application so that students will keep questioning the sample programs". (2) Flowcharts illustrated the sequence of logical operations. Student 1 said, "Flowchart is useful. It will let us to understand the procedure of how the program works", Student 5 said, "Like flowchart, it helps me because you know how the computer thinks in that way" and Student 7 said, "The flowchart can enhance the clarification of some complicated theories". (3) Animations visualized programming concepts. Other than flowcharts, two students explained how an animated apple tree (in the lecture notes) helped them to learn (see Figure 1). Student 3 said, "Yes, the apple thing. I felt useful… Easier to understand when the concept that you feel complicated" and Student 6 said, "It's better to use something like apple tree because everyone should know what is an apple tree, but they do not know what is a *for* loop or *while* loop. I think that is a very good diagram". According to the

Table 3. Students' responses to the usefulness of flipped materials

Learning Materials	Disagree %	Neutral %	Agree %
Video lectures	16	34	50
Lecture notes	1	0	99
- Sample programs in the lecture notes	3	11	86
- Diagrams in the lecture notes	2	11	87
Tutorial questions	1	3	96
Online quizzes	2	7	91

Figure 1. An animated apple tree to demonstrate a "for" loop structure



students, flowcharts and animated diagrams should be used to illustrate the abstract and complicated programming concepts such as the repetition structures.

More than 90% of the students used tutorial questions and online quizzes to learn. The students believed in practice made perfect. Student 1 said, "I will do the tutorial myself for exam purpose", Student 3 said, "Tutorial allows me train and I will more aware that what is the common mistake that I will make during the exam" and Student 7 said "I study the tutorial and online quiz while seeing the answers before tests". According to the students, tutorial questions and online quizzes were helpful in preparing them for exams.

50% of the students used video lectures to learn. There were two groups of students. (1) Watched the video lectures. Student 4 said, "When we don't know how to do, we always refer back to the videos… You can check back where the mistakes are and can avoid it." and Student 5 said, "Video is best because you can always look back how the lecturer did it. Compared to pictures, video is still the best… you can listen and not like 2-D in pictures. Like from this step to another step, there are minor changes, you can always look back at the videos. But, in pictures, you may not notice it". Apparently, video lectures were useful in explaining certain programming concepts that could not be fully explained in the lecture notes. Video lectures could help students to understand better, especially the slow learners and students with learning difficulties. (2) Did not watch the video lectures. Student 1 said, "I normally won't use video except the concept is really complicated, I will watch it" and Student 3 said, "I didn't watch any because I understand". Despite knowing the benefits of video lectures, some students would only watch the video lectures if needed.

RQ4: What is the Relationship Between Various Flipped Materials and Students' Test Performance?

There were significant correlations between certain flipped materials and test performance (see Table 4). Lecture notes and tutorial questions were weakly positive correlated with test performance ($r_s = .225$; $r_s = .223$, $p < .01$). The finding indicated that students who used lecture notes and tutorial questions to learn were more likely to perform better in the test. There were no significant correlations between test performance, and video lectures or online quizzes, respectively. Correlation analysis suggested that students should use lecture notes and tutorial questions to learn programming.

RQ5: Do Gender Differences Affect Students' Test Performance?

In the study sample, 84% were males and 16% were females. The Shapiro-Wilk test showed that the dependent variable (test performance) significantly deviated from a normal distribution ($p < .05$). Thus, a Mann-Whitney test was performed (see Table 5). Based on the measures of central tendency, females performed better than males in the test (F: $\bar{x} = 83.36$, $\tilde{x} = 84.98$; M: $\bar{x} = 76.93$, $\tilde{x} = 77.97$). The Mann-Whitney test further confirmed the result ($U = 2251.5$, $p < .05$, $r = .155$). The p-value (.025) indicated the rejection of the null hypothesis of equal medians at 5% significance level. It was a surprising finding given that more males (26%) had prior programming experience compared to females (17%). This implied that the late exposure of females to computer programming did not result in lower test performance. The findings were supported by two past studies (Burnett et al., 2010; Murphy et al., 2006).

Table 4. Correlations of various flipped materials and test performance

	Video Lectures	Lecture Notes	Tutorial Questions	Online Quizzes
Correlation Coefficient	-0.122	.225**	.223**	0.023
Sig. (2-tailed)	0.079	0.001	0.001	0.741
*. Correlation is significant at the 0.05 level (2-tailed). **. Correlation is significant at the 0.01 level (2-tailed).				

Table 5. Results of the Mann Whitney U-test for gender differences

	Males (M)	Females (F)	Mann-Whitney U	Z	Asymp. Sig. (2-tailed)
TEST PERFORMANCE			*2251.500*	*-2.242*	*.025**
Mean Rank	100.87	126.28			
Sum of Ranks	17651.50	4293.50			
Mean ($\overline{x}$)	76.93	83.36			
Median ($\tilde{x}$)	77.97	84.98			
FLIPPED SESSIONS					
Pre-class self-study			2415.500	-1.804	.071
Mean Rank	101.50	121.46			
Sum of Ranks	17815.50	4129.50			
In-class lecture			2796.500	-.623	.534
Mean Rank	103.98	110.25			
Sum of Ranks	18196.50	3748.50			
In-class tutorial			2869.500	-0367	.714
Mean Rank	104.40	108.10			
Sum of Ranks	18269.50	3675.50			
Peer learning			2713.500	-.890	.373
Mean Rank	103.51	112.69			
Sum of Ranks	18113.50	3831.50			
FLIPPED MATERIALS					
Video lectures			*2364.500*	*-1.966*	*.049**
Mean Rank	101.51	122.96			
Sum of Ranks	17764.50	4180.50			
Lecture notes			2568.500	-1.468	.142
Mean Rank	102.68	116.96			
Sum of Ranks	17968.50	3976.50			
Tutorial questions			2564.500	-1.446	.148
Mean Rank	102.65	117.07			
Sum of Ranks	17964.50	3980.50			
Online quizzes			2495.00	-1.655	.098
Mean Rank	102.26	119.12			
Sum of Ranks	17895.00	4050.00			
* significant at 0.05 level (2-tailed)					

In a flipped programming classroom, females used video lectures more frequently than their male counterparts ($U = 2364.5$, $p < .05$, $r = .136$). The p-value of .049 indicated that the Mann-Whitney test rejected the null hypothesis of equal medians at the 5% significance level. However, there were no significant differences between males and females in various flipped sessions and other flipped materials such as lecture notes, tutorial questions and online quizzes. The p-values of .071 (pre-class self-study), .534 (in-class lecture), .714 (in-class tutorial), .373 (peer learning), .142 (lecture notes), .148 (tutorial questions) and .098 (online quizzes) indicated that there was insufficient evidence to reject the null hypotheses at the 5% significance level. According to Hao (2016), there were no significant gender differences in flipped learning readiness. However, the present study indicated that female students valued more the benefits of learning with video lectures.

Correlation analysis revealed that flipped learning affected males' and females' test performance differently (see Table 6). For males, test performance was weakly positive correlated with in-class lecture ($r_s = .181$, $p < .05$), in-class tutorial ($r_s = .196$, $p < .01$), lecture notes ($r_s = .173$, $p < .05$) and tutorial questions ($r_s = .209$, $p < .01$). On the contrary, test performance was weakly negative correlated with video lectures ($r_s = -.166$, $p < .05$). As for females, there was a moderately positive correlation between test performance and lecture notes ($r_s = .477$, $p < .01$) and a moderately negative correlation between test performance and peer learning ($r_s = -.456$, $p < .01$). The results indicated that males could improve their programming proficiency through in-class lecture and tutorial sessions (using the lecture notes and tutorial questions). However, video lectures had a negative impact on males' test performance. As for females, lecture notes could help them to improve programming skills. However, peer learning should be minimized as it has an adverse effect on females' test performance.

RQ6: Do Programming Background Differences Affect Students' Test Performance?

In the sample of study, most of the students (80%) had no prior programming experience. A Mann-Whitney test was used to compare the differences of students with (prog.) and without prior programming experience (no prog.) (see Table 7). Based on the measures of central tendency, students without prior programming experience seemed to perform better in the test (no prog: $\bar{x} = 78.34$, $\tilde{x} = 79.01$; prog: $\bar{x} = 76.02$, $\tilde{x} = 77.97$). However, the Mann-Whitney test had negated this argument ($U = 2922.5$, $p > .05$, $r = .104$). The p-value (.133) indicated that there was not enough evidence to reject the null hypothesis at the 5% significance level. This finding was supported by Murphy et al.

Table 6. Correlations of various flipped sessions/materials and test performance (gender differences)

	Males		Females	
	r_s	Sig. (2-tailed)	r_s	Sig. (2-tailed)
FLIPPED SESSIONS				
Pre-class self-study	.058	.443	.113	.526
In-class lecture	.181*	.017	.252	.150
In-class tutorial	.196**	.009	.143	.419
Peer Learning	-.139	.066	-.456**	.007
FLIPPED MATERIALS				
Video lectures	-.166*	.028	.018	.920
Lecture notes	.173*	.022	.477**	.004
Tutorial questions	.209**	.006	.256	.144
Online quizzes	-.030	.691	.280	.109

r_s – correlation coefficient
*. Correlation is significant at the 0.05 level (2-tailed).
**. Correlation is significant at the 0.01 level (2-tailed).

Table 7. Results of the Mann Whitney U-test for programming background differences

	Prog.	No Prog.	Mann-Whitney U	Z	Asymp. Sig. (2-tailed)
TEST PERFORMANCE			2922.500	-1.502	.133
Mean Rank	92.28	108.10			
Sum of Ranks	3783.50	18161.50			
Mean ($\overline{x}$)	76.02	78.34			
Median ($\tilde{x}$)	77.97	79.01			
FLIPPED SESSIONS					
Pre-class self-study			3329.000	-.345	.730
Mean Rank	102.20	105.68			
Sum of Ranks	4190.00	17755.00			
In-class lecture			3414.000	-.097	.923
Mean Rank	105.73	104.82			
Sum of Ranks	4335.00	17610.00			
In-class tutorial			3122.500	-1.040	.298
Mean Rank	112.84	103.09			
Sum of Ranks	4626.50	17318.50			
Peer learning			2823.500	-1.963	.050
Mean Rank	89.97	108.69			
Sum of Ranks	3684.50	18260.50			
FLIPPED MATERIALS					
Video lectures			3177.500	-.798	.425
Mean Rank	98.50	106.59			
Sum of Ranks	4038.50	17906.50			
Lecture notes			3205.500	-.801	.423
Mean Rank	110.82	103.58			
Sum of Ranks	4543.50	17401.50			
Tutorial questions			3433.000	-.036	.971
Mean Rank	105.27	104.93			
Sum of Ranks	4316.00	17629.00			
Online quizzes			3386.000	-.186	.853
Mean Rank	103.59	105.35			
Sum of Ranks	4247.00	17698.00			

Prog. - with prior programming experience
No Prog. - without prior programming experience

(2006), who reported no correlation between previous programming experience and programming achievement.

Further analysis revealed that there were no significant differences between students with and without prior programming experience in various flipped sessions and the usage of flipped materials. The p-values of .730 (pre-class self-study), .923 (in-class lecture), .298 (in-class tutorial), .050 (peer learning), .425 (video lectures, .423 (lecture notes), .971 (tutorial questions) and .853 (online quizzes) indicated that there was insufficient evidence to reject the null hypotheses at the 5% significance level. This suggested that all students regardless of programming background, responded equally to a flipped classroom approach.

Correlation analysis revealed that flipped learning affected the test performance of different groups of students (with or without programming background) (see Table 8). For students with prior programming experience, there was a moderately positive correlation between test performance and in-class tutorial ($r_s = .396$, $p < .05$), lecture notes ($r_s = .484$, $p < .01$) and tutorial questions ($r_s = .503$, $p < .05$), respectively. For students without prior programming experience, test performance was weakly positive correlated with pre-class self-study ($r_s = .164$, $p < .05$), in-class lecture ($r_s = .171$, $p < .05$) and lecture notes ($r_s = .172$, $p < .05$). However, test performance was weakly negative correlated with peer learning ($r_s = -.155$, $p < .05$). The results indicated that lecture notes were important for both groups of students. However, students with prior programming experience should attend more tutorial sessions. As for those without prior programming experience, they should do self-preparation before attending a class and attend all the lecture sessions. Since they are novice learners, peer learning should be minimized to avoid confusion and misunderstanding of concepts learned.

Table 8. Correlations of various flipped sessions/materials and test performance (programming background differences)

	Prog.		No Prog.	
	r_s	Sig. (2-tailed)	r_s	Sig. (2-tailed)
FLIPPED SESSIONS				
Pre-class self-study	-.197	.217	.164*	.034
In-class lecture	.257	.105	.171*	.027
In-class tutorial	.396*	.010	.138	.074
Peer Learning	-.286	.069	-.155*	.046
FLIPPED MATERIALS				
Video lectures	-.100	.534	-.139	.073
Lecture notes	.484**	.001	.172*	.026
Tutorial questions	.503*	.001	.151	.050
Online quizzes	.214	.180	-.036	.645

r_s – correlation coefficient
*. Correlation is significant at the 0.05 level (2-tailed).
**. Correlation is significant at the 0.01 level (2-tailed).

CONCLUSION

The results presented this study suggest that students are not fully ready for a flipped classroom. Most of the students would still prefer the face-to-face lectures and tutorials (in-class activities). The finding coincides with Baldwin (2015) and Piteira and Costa (2013). During the in-class lecture, students could get a better lecture explanation, troubleshooting knowledge and answers to various queries. During the in-class tutorial, students can use their knowledge and creativity to solve hands-on programming problems. The finding concurs with Piteira and Costa (2013), in which they have reported that students prefer in-class practice and exercise sessions. In-class activities are important because they have a positive impact on students' test performance, especially for the male students. Students without prior programming experience will benefit more from lectures while students with prior programming experience will benefit more from tutorial sessions. More than three-quarters of the students like to learn from peers. However, this approach induces a negative impact on students' test performance, especially among the female students. Apparently, the disadvantages (confuse each other) overtake the advantages (learn from each other). About half of the students doing self-study and are prepared before going to class. Unfortunately, pre-class self-study has no impact on students' test performance, except for those without prior programming experience. This means, novice learners should best prepare themselves before classes. This finding contradicts with Dazo et al. (2016). The contradiction could be explained by students' preferring post-class self-study (revision purposes) rather than pre-class self-study (study preparation).

Students generally find all flipped materials useful except the video lectures. This finding coincides with Piteira and Costa (2013). Lecture notes are the main sources of reference. Students who learn from lecture notes could perform better in the tests, regardless of gender and programming background. Tutorial questions are important too. The questions allow students to practice and prepare themselves for the test. Students who learn from the tutorial questions could perform better in the tests, especially the male students and those with prior programming experience. Students find online quizzes useful, but the materials have no impact on students' test performance. The least popular flipped materials are video lectures. Females like to learn from video lectures. However, video lectures have no impact on students' test performance, except for male students (negative impact). Generally, slow learners can benefit more from the video lectures.

Despite the unpopularity of pre-class self-study session and video lectures, the findings of this study have revealed the direction of future studies – variability of flipped learning and differences in gender and programming background. For instance, students regardless of programming background, respond equally to a flipped classroom approach. Therefore, flipped learning could be an appropriate classroom model for students coming from diverse educational background. Additionally, the late exposure of females to computer programming does not result in lower test performance. In fact, females may outperform males given the right learning environment.

ACKNOWLEDGMENT

This research was funded by the University of Nottingham Malaysia Teaching and Learning Grant (2018).

REFERENCES

Akçayır, G., & Akçayır, M. (2018). The flipped classroom: A review of its advantages and challenges. *Computers & Education*, *126*, 334–345. doi:10.1016/j.compedu.2018.07.021

Awidi, I. T., & Paynter, M. (2019). The impact of a flipped classroom approach on student learning experience. *Computers & Education*, *128*, 269–283. doi:10.1016/j.compedu.2018.09.013

Baldwin, D. (2015). Can we "flip" non-major programming courses yet? In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. Kansas City, MO: ACM. doi:10.1145/2676723.2677271

Barbour, C., & Schuessler, J. B. (2019). A preliminary framework to guide implementation of the flipped classroom method in nursing education. *Nurse Education in Practice*, *34*, 36–42. doi:10.1016/j.nepr.2018.11.001 PMID:30439682

Burnett, M., Fleming, S. D., Iqbal, S., Venolia, G., Rajaram, V., Farooq, U., … Czerwinski, M. (2010). Gender differences and programming environments: Across programming populations. In *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '10).* Bolzano-Bozen, Italy: ACM. doi:10.1145/1852786.1852824

Cohen, L., Manion, L., & Morrison, K. (2007). *Research Methods in Education* (6th ed.). Routledge. doi:10.4324/9780203029053

Dazo, S. L., Stepanek, N. R., Fulkerson, R., & Dorn, B. (2016). An empirical analysis of video viewing behaviors in flipped CS1 courses. *ACM Inroads*, *7*(4), 99–105. doi:10.1145/3007625

Giannakos, M. N., Krogstie, J., & Chrisochoides, N. (2014). Reviewing the flipped classroom research: Reflections for computer science education. In *Proceedings of the Computer Science Education Research Conference (CSERC '14).* Berlin, Germany: ACM. doi:10.1145/2691352.2691354

Hao, Y. W. (2016). Middle school students' flipped learning readiness in foreign language classrooms: Exploring its relationship with personal characteristics and individual circumstances. *Computers in Human Behavior*, *59*, 295–303. doi:10.1016/j.chb.2016.01.031

Isomöttönen, V., & Tirronen, V. (2016). Flipping and blending - An action research project on improving a functional programming course. *ACM Transactions on Computing Education*, *17*(1), 1–35. doi:10.1145/2934697

Jenkins, T. (2002). On the difficulty of learning to program. In *Proceedings of the 3rd annual LTSN-ICS conference*. Loughborough, UK: LTSN Centre for Information and Computer Sciences.

Johnson, B., & Christensen, L. (2008). *Educational Research Quantitative, Qualitative and Mixed Approaches*. Sage Publications.

Kim, J. A., & Kim, H. J. (2017). Flipped learning of Scratch programming with code.org. In *Proceedings of the 2017 9th International Conference on Education Technology and Computers (ICETC 2017).* Barcelona, Spain: ACM. doi:10.1145/3175536.3175542

Köppe, C., & Bartilla, A. (2017). Towards a pattern language for increasing gender diversity in computer science education. In *Proceedings of the VikingPLoP 2017 Conference on Pattern Languages of Program (VikingPLoP '17).* ACM. doi:10.1145/3158491.3158501

Lacher, L. L., & Lewis, M. C. (2015). The effectiveness of video quizzes in a flipped class. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. Kansas City, MO: ACM. doi:10.1145/2676723.2677302

Lopes, A. P., & Soares, F. (2018). Perception and performance in a flipped financial mathematics classroom. *International Journal of Management Education*, *16*(1), 105–113. doi:10.1016/j.ijme.2018.01.001

Maragos, K., & Grigoriadou, M. (2007). Designing an educational online multiplayer game for learning programming. In *Proceedings of the Informatics Education Europe II Conference (IEEII 2007).* Thessaloniki, Greece: South-East European Research Center.

Marsh, C. J. (2010). A sub-saharan comparative study of university students' attitudes towards computer programming. In *Proceedings of the 15th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE' 10)*. Bilkent, Ankara, Turkey: ACM. doi:10.1145/1822090.1822102

Murphy, L., Richards, B., McCauley, R., Morrison, B. B., Westbrook, S., & Fossum, T. (2006). Women catch up: Gender differences in learning programming concepts. In *Proceedings of the 37th SIGCSE technical symposium on Computer science education (SIGCSE '06)*. Houston, TX: ACM. doi:10.1145/1121341.1121350

Ng, E. M. W. (2018). Integrating self-regulation principles with flipped classroom pedagogy for first year university students. *Computers & Education*, *126*, 65–74. doi:10.1016/j.compedu.2018.07.002

Njie-Carr, V. P. S., Ludeman, E., Lee, M. C., Dordunoo, D., Trocky, N. M., & Jenkins, L. S. (2017). An integrative review of flipped classroom teaching models in nursing education. *Journal of Professional Nursing*, *33*(2), 133–144. doi:10.1016/j.profnurs.2016.07.001 PMID:28363388

Piteira, M., & Costa, C. (2013). Learning computer programming: Study of difficulties in learning programming. In *Proceedings of the 2013 International Conference on Information Systems and Design of Communication (ISDOC '13)*. Lisboa, Portugal: ACM. doi:10.1145/2503859.2503871

Tan, C., Yue, W. G., & Fu, Y. (2017). Effectiveness of flipped classrooms in nursing education: Systematic review and meta-analysis. *Clinical Nursing Research*, *4*(4), 192–200.

Wagner, I. (2016). Gender and performance in computer science. *ACM Transactions on Computing Education*, *16*(3), 1–16. doi:10.1145/2920173

Su-Ting Yong is an Associate Professor in the Department of Foundation in Engineering, The University of Nottingham Malaysia. She joined the University in 2008, having taught in a few universities for a number of years. Dr Yong obtained her Bachelor's Degree in Science and Computer with Education (Mathematics) and Master's Degree in Information Technology from The University of Technology Malaysia. She completed her PhD in Engineering Education at The University of Nottingham. She is a Senior Fellow of The Higher Education Academy and has a wide variety of research interests, largely focused on technology in mathematics education, educational games, gamification and programming. Her latest project is funded by the University Teaching and Learning Fund.

K. M. Tiong is a Certified Analytics Professional and has more than 15 years of experience in higher education, teaching at foundation and undergraduate levels. Currently, he is an Assistant Professor at The University of Nottingham Malaysia. He trained as a data scientist at The Center of Applied Data Science (CADS). He has also authored an educational guidebook titled "Decide Wisely".

Andy Chan is the current Dean of Science and Engineering at the University of Nottingham Malaysia. His responsibility is to oversee all research, teaching, external and operation matters of the faculty. He is a member of the UNMC management board. Professor Chan's area of expertise is on air pollution and computational fluid dynamics, especially towards urban climates. During the 2003 severe acute respiratory syndrome (SARS) episode, he was one of the team leaders appointed by the World Health Organisation (WHO) to discover the dispersion of mechanisms of the SARS virus in the dense cityscapes of Hong Kong. He is currently the leader of the multinational 7SEASNASA project (Seven South-east Asian Studies-National Aeronautics and Space Administration), which studies the propagation of haze in Southeast Asia. He is currently also serving as the government expert consultant on the development of urban planning guidelines against air pollution and extreme weather conditions.

Khiew Poi Sim (PhD) is a material physicist specializing in nanoscience technology and advanced materials. He has been involved intensively in many scientific and technology based research projects that are sponsored by a number of reputable funding agencies, such as MOSTI, MOHE, MOA and CRADLE. He has published more than 100 papers in international peer-reviewed journals and conference proceedings. His inventions have been recognized by a series of innovation awards for the outstanding accomplishments in the advanced materials research. His current research work focuses on nanoscience and nanotechnology advancement, especially on utilizing the nanostructures (semiconductor, ceramic and polymer composite) for photocatalysis, biosensing, organic photovoltaic and electrochemical energy storage applications.