Journal of University Teaching & Learning Practice

Volume 16 | Issue 5 Article 4

2019

An Investigation of Students' Learning Achievement and Perception using Flipped Classroom in an Introductory Programming course: A Case Study of Thailand Higher Education

Jarutas Pattanaphanchai
Prince of Songkla University, jarutas.p@psu.ac.th

Recommended Citation
Pattanaphanchai, Jarutas, An Investigation of Students' Learning Achievement and Perception using Flipped Classroom in an Introductory Programming course: A Case Study of Thailand Higher Education, *Journal of University Teaching & Learning Practice*, 16(5), 2019.
Available at:https://ro.uow.edu.au/jutlp/vol16/iss5/4

An Investigation of Students' Learning Achievement and Perception using Flipped Classroom in an Introductory Programming course: A Case Study of Thailand Higher Education

Abstract

Computer programming is a difficult subject for most novice programming students which leads to a high rate of dropout or failure. Flipped classrooms have been increasingly used to teach programming students to practice their programming skills in the class based on their knowledge acquired from outside the classroom. This study investigated students' learning achievement which was measured by their performance on a coding test and an exam. We compared students' programming test scores of the traditional classroom in semester 1 of the academic year 2016 with the flipped classroom in semester 1 in 2017. In addition, we studied students' perception of flipped classrooms in a structured programming course. The population of this study consisted of 69 second-year students of the Information and Communication Technology programme in the Faculty of Science, Prince of Songkla University, Thailand. The research instrument for investigating students' perception of flipped classrooms was a questionnaire. Students were positive in their perception of flipped classrooms and agreed that in-class activities improved their understanding of concepts. Moreover, students in the flipped classroom environment had a significantly higher test score compared to students in a traditional lecture class.

Keywords

flipped classroom, learning achievement, programming

Cover Page Footnote

This study funded by Research Funding of the year 2018 from Faculty of Science, Prince of Songkla University and Prince of Songkla University.

Introduction

Teaching programming for novice students is a challenging task, as it needs to apply concepts of programming into hands-on coding. The traditional style of teaching involves lectures, programming assignments and written examinations. Most students struggle to understand and implement programming concepts resulting in high failure rates (Butler & Morgan 2007). A number of alternative teaching methods have been introduced and explored to improve students' programming. The flipped classroom, which is gaining popularity in higher education, is a pedagogical approach that moves direct instruction in the learning content from group space in the classroom to the individual learning space. In the students' individual space, they contact the new material in form of structured activities, which are provided through guided instructions in either a hard-copy book or digitally (in an electronic book, video or tutorial via an online system). The instructions contain a mixture of text, pictures and diagrams. In the group spaces, students apply their understanding to problems and work together, with either the instructor or their classmates (Fulton 2012, Bergmann & Aaron 2012, Talbert 2017). The flipped-learning model consists of two main characteristics. The first is direct instruction; the lecturer provides guidelines to students on how they are expected to encounter the new materials. The guidelines express the structure and sequence of learning the new matter. Direct instruction can be in several forms; for example, the lecturer can assign activities that students need to summarise or ask them to draw a mind map after reading the material to prove their understanding of the content. The second characteristic is an intentional approach to get students engaged in teams. The engagement involves dynamic and creative activities that support learning. Activities in class can include peer-assisted learning, cooperative learning, problem-based learning, group presentations, a think-pair-exercise discussion or classroom responses to provoking questions using technology such as clickers (Bishop & Verleger 2013, Hawks 2014, Talbert 2017). The advantages claimed for the flipped classroom include improved grades, self-paced learning, increased engagement in learning and improved self-efficacy in independent learning (Enfield 2013, Herreid & Schiller 2013, Gilboy, Heinerichs & Pazzaglia 2015).

The Information and Communication Technology (ICT) Programme in the Faculty of Science, Prince of Songkla University (PSU), Thailand, provides an introductory structured programming course for sophomore-level students in their first semester. In 2016, the introductory structured programming course had been conducted in traditional teaching style in which the instructor delivered subject content by giving a lecture. Then, students practiced their understanding of the content via hands-on mandatory programming assignments and programming laboratories. The course materials (PowerPoint slides, links to book chapters, laboratory instruction sheets and course information) were provided via the university's e-learning system (LMS2, an implementation based on Moodle) (Dougiamas & Taylor 2003, Cole & Foster 2007). Despite the students having easy access to material, the high failure rate and lack of programming proficiency became clear at the end of the semester. The researchers, therefore, implemented a flipped classroom in the expectation of improved students' learning experiences and learning achievement. The purpose of this research was to study students' satisfaction with the flipped-classroom approach and students' satisfaction with the in-class activities in the flipped classroom, and to compare students' learning achievement between the conventional classroom in semester 1 of the academic year 2016 and the flipped classroom in semester 1 in 2017. The hypotheses tested were:

1. There will be no significant differences in pre- and post-mean ratings of students' understanding of the flipped classroom.
2. There will be no significant differences in mean ratings of satisfaction with in-class activities in the flipped classroom and the neutral rating.

3. There will be no significant difference in programming examination scores between the semester 1/2016 and the semester 1/2017 cohorts.

The benefits of this study are that it may help lecturers design their teaching of introductory structured programming in more effective ways and that students may improve their ability to understand and implement structured programming.

Literature review

A number of studies have reported the implementation of flipped classrooms in domains ranging from medical to business and programming. O'Flaherty and Phillips reviewed relevant research regarding the application of flipped-classroom pedagogy, finding that the flipped-classroom approach improved students' academic performance and increased student and staff satisfaction with the learning process (O'Flaherty & Phillips 2015). In addition, the study of the effects of flipped classrooms in four different computer-science classes – rapid prototyping and interface building, human-computer interaction, web applications development and introduction to media computation – were investigated. The research proposed a set of flexible strategies to construct flipped classrooms in a computer-science domain: out-of-class preparation, using or creating online videos and preparing in-class activities to scaffold skills development and identify students' misconceptions (Maher, Latulipe, Lipford & Rorrer 2015). Prober and Khan proposed a flipped-classroom learning model for physicians in which students accessed 10 minutes of online video to learn a new concept at their own pace, then participate in interactive sessions where they applied their new knowledge. Their model emphasised the need for a newly designed curriculum for medical students to increase the level of understanding and skill imparted in medical education (Prober & Khan 2013). Tune, Sturek and Basile compared the effectiveness of a traditional lecture-based approach and a flipped classroom for the cardiovascular, respiratory and renal physiology course to first-year postgraduate students. Both groups received the same notes and recorded lectures. In the implementation of the flipped classroom, students watched the prerecorded lectures before class and attended the class for practicing with a quiz or homework. The traditional lecture-based group was exposed to the content in class and there were no quizzes. The students from both groups took the same multiple-choice exams. The results showed that the students in the flipped classroom gained higher scores than the students in the traditional class (Tune, Sturek & Basile 2013).

Amresh, Carberry and Remiani implemented the flipped-classroom method to teach introductory programming to engineering, engineering-technology and software-engineering undergraduates. The students were divided into three sections: two sections studied in a flipped classroom (experimental) and the third in a traditional class (control). The experimental sections were counterbalanced: one section studied in the flipped classroom for the first half of the semester and then in a traditional class for the remainder of the semester, while this was reversed for the other experimental section. The control section studied in a traditional class for the whole semester. The results showed that the flipped classroom was correlated with improved student grades. In addition, students' computing self-efficacy improved (Amresh, Carberry & Femiani 2013). Campbell, Horton, Craig and Gries explored the flipped-classroom method to teach an introductory programming course. For the flipped-classroom implementation, students prepared for their classes by watching short lecture videos and answering questions on in-video quizzes. In the class time, the students worked through exercises facilitated by the instructor and teaching assistant. The results showed that students' enthusiasm for and enjoyment of the course increased over the semester (Campbell, Horton, Craig & Gries 2014). Findlay-Thompson and Mombourquette studied academic outcomes of a flipped classroom in a business course, finding no grade differences between students studying in the flipped classroom and those in traditional-lecture style class. However, the study showed a positive learning environment in the flipped classroom because students felt that they had time to address questions to their lecturer or fellow students (Findlay-Thompson & Mombourquette 2014). Kim, Kim, Khera and Getman proposed nine design principles for the flipped classroom and examined three instances of the flipped classroom across disciplines. The results from that short survey showed that students were satisfied with the flipped-classroom activities because it helped them understand the course concepts (Kim, Kim, Khera & Getman 2014). Jonsson combined a flipped-classroom approach with some peer discussion and just-in-time teaching in a course on object-oriented programming. Of the students experiencing this experimental condition, 81% passed the course, compared to 60% of the students experiencing the control condition of a traditional class. Moreover, 58% of the students in the experimental condition passed with a good grade compared to 32% in the control condition (Jonsson 2015). Alhazbi investigated the effectiveness of using the flipped classroom to teach computer programming. The researcher conducted a pilot study using one topic in the course at the end of the semester. The content of the topic was recorded as a lecture video and uploaded on the university's learning system two days before the class time. The online quizzes were provided for the students to answer after watching the video. Students solved problems individually and as a group during class. They then participated in a discussion to summarise the correct solution. The results showed that students had positive attitudes towards the flipped classroom, and that the syntax and structure of their programming language improved (Alhazbi 2016). Moreover, Turan and Göktaş studied the impact of the flipped-classroom method on students' motivation in an introductory computer course. The study divided the students into two groups, both of which were taught the same content: the control group was taught with the lecture-based method and the experimental group in a flipped-classroom approach. In the flipped classroom, the students watched a video before coming to the class, and did their homework during the class time. The results showed that students who studied in the flipped classroom were more motivated than students who studied in the traditional class. Students' motivation was increased by hands-on in-class activities, group work and gamification activities (Turan & Göktaş 2018).

Despite the positive results from several studies on the attitudes and perceptions of students who experienced flipped classes in programming courses in a variety of contexts, few studies have been reported in Thailand. Puarungroj investigated the perception of students towards flipped classrooms in a programming course, providing online video lectures and an online book. In each class, students first answered a 10-minute quiz, then worked in twos or threes on practice activities. The results showed that students were highly satisfied with their interaction with the instructor and were positively engaged in their classes. Students were mostly satisfied with learning at their own pace using online materials (Puarungroj 2015). However, the study did not investigate the learning output. In this study, we investigated both students' opinions towards flipped classrooms in an introductory programming course and students' learning results in term of their testing scores in a coding examination.

Method

This was a quantitative study using a questionnaire as the research instrument to collect data. The study was conducted in the first semester of the 2017 academic year in a sophomore-level course on structured programming. The details are presented below.

Participants

Sixty-nine students were enrolled in the course. For most, this was their first exposure to programming.

Course design

To implement the flipped-classroom pedagogy, the author prepared materials that students could use to learn in their individual space and learning activities for use in the group space. The learning materials consisted of course content in the form of PowerPoint slides, video lectures and video tutorials. In addition, the author designed an instruction guide for the students to use with the PowerPoint slides or the videos. The sequence of learning materials was designed to follow the steps of basic knowledge in structured programming such as data types and input and output commands, and then moving to more complex topics such as control statements, arrays, pointers, functions and file operations. The video material was divided into two categories: self-produced by the author and curated tutorials from various online resources. Five-minute video tutorials for using Codeblock (The Code::Blocks Team 2019), the programming editor used in the course, and for compiling and running programs were recorded using the free version of CamStudio Recorder (Rendersoft Development 2019). The online videos curated from online resources were selected based on three criteria: their content, length and production style. The selected videos needed to deliver content that was consistent with or almost identical to what would have been delivered in the class with the PowerPoint slides, with a length of six to nine minutes. In addition, the picture-to-picture style was chosen, in which an instructor talks with a head or half-body overlay on top of the delivered content, with either screencasts or coding demonstrations. Research has shown that this length of video and presenting style retain students' engagement so that they tend to watch the whole video (Guo, Kim, & Rubin, 2014). These tutorial and lecture videos were uploaded to the LMS2 system (Beatty & Ulasewicz, 2006; Bremer & Bryant, 2005). Three to six lecture videos were provided each week depending on the topic. For each topic in the course, self-check quizzes composed of four multiple-choice or fill-in-the-blank questions were given as pre-tests and post-tests. The questions were basic, based on the content that students needed to understand. Students were allowed to practice a quiz as many times as they desired, with immediate feedback. The learning activities were designed to get students' engagement in the class via several types of activities: pair or group discussions, question-based problem-solving in a group, the demonstration of programming concepts using daily life objects and classroom responses with provoking questions by using technology such as Kahoot!, a student-response system using game-like designs for quizzes, discussions and surveys. Students could access the quiz or survey through any device with a web browser or its application. This type of game-based learning promotes students' engagement in the class while they learn through the game (Dellos 2015, Graham 2015).

Course assessment

The course included both summative and formative assessments. The summative assessment evaluated students' programming skills, involving three mini-tests, a midterm examination and a final examination. The mini-tests were at weeks 5, 9 and 13 of the course. The midterm examination was at week 10 and the final examination at week 17. All programming tests were timed, and contained a finite number of problems specific to the course. Students accessed the LMS2 and wrote code to solve the assigned problems, which were randomly selected from a question bank. The midterm and final examinations assessed students' understanding of programming and covered the topics of the first and second halves of the course respectively.

The formative evaluation was conducted weekly to measure students' understanding of each topic using Kahoot!. The quiz was composed of five multiple-choice questions. The scores from the quizzes did not directly contribute to the students' overall grade, but the quality of interaction during class time, as displayed by either answering quizzes, participating in group discussions or completing coding exercises, was taken into consideration for a participation component of 5%.

Research design

This study investigated students' perceptions and learning output of a flipped classroom in a structured programming course in the second year. The course duration was 15 weeks, with two additional weeks for midterm and final examinations.

The first hour of the first lesson was used to evaluate students' prior knowledge about the flipped classroom using a self-rating questionnaire. The self-rating session was conducted at the same standard as the examination session itself, where students were asked to answer an online questionnaire in the computer laboratory. Each student was provided with a computer that they could use to access the online questionnaire. The questions asked about their understanding of the flipped-classroom teaching method, level of programming skill and familiarity with the learning-system tool. In addition, students' grades in three basic first-year courses (mathematics, physics and chemistry) were used to assess their level of computational skill. In the second hour, the author explained the flipped-classroom approach so that students would know what to expect. In addition, the author articulated expectations about class attendance and pre-class preparation to students by instructing them to study and watch the assigned videos and attempt the pre-test and post-test quizzes provided on LMS2 to assess their understanding of the topic prior to attending the class. The students were also urged to note down any questions that they might have when watching the videos. The pre-test and post-test evaluation were also used as an indicator of students' preparation. The students were warned that failing to prepare for class would be penalised by lower marks for class participation.

Starting in the second week, the first 15-20 minutes of each class were reserved for assessing students' understanding or clarifying the content covered in the videos. The students then participated in a wide range of hands-on tasks, such as pair or group discussions, question-based problem-solving in a group and the demonstration of programming concepts using daily life objects. One example of question-based problem-solving in a group would be solving problems using control structure statements in the C programming language. An example for demonstrating programming concepts using daily life objects would be using a plastic box to represent the elements of the array and demonstrate adding elements to and removing elements from an array variable. The approximate time spent on each activity was about 65 to 70 minutes. At the end of the class, students completed a 10- to 15-minute post-test designed to assess their understanding of the class topic. During the last week of the semester, students repeated the self-rating questionnaire, which had an added section asking about students' satisfaction with the in-class activities.

Research instruments for data collection

The author used LMS2 activities log to collect students' participation in pre-test and post-test activities during their self-learning time. In addition, a self-report questionnaire was developed to collect students' opinions about flipped-classroom teaching methods and their satisfaction with the in-class activities. The self-reported questionnaire was developed according to the following method (Crawford & Kelder 2019). First, the author reviewed documents, theories and relevant studies from different sources. Then, the author identified the elements that could be used to evaluate the students' opinions and satisfaction. A three-section questionnaire was then developed based on these elements and the study objectives. The first section asked the students' gender and the number of times they attended classes. The second section included six questions about their understanding of flipped classrooms, their IT skills and their level of programming knowledge. The last section consisted of open-ended questions that elicited opinions and comments. The version of the questionnaire administered in week 15 also included a fourth section with 15 extra questions asking students to rate their satisfaction towards the flipped classroom, teaching methods and the in-class activities using a five-point Likert scale from “totally disagree” to “totally agree”. The examination of the quality of the research instrument in terms of content validity was done using the index of item-objective congruence (ICO) (Rovinelli & Hambleton 1977). Three experts in educational technology were asked to evaluate each item in the questionnaire and score it as +1 (congruent), 0 (questionable) or -1 (incongruent). Items that scored lower than 0.5 were revised, and those that scored 0.5 or more remained in the questionnaire unchanged. The reliability of the questionnaire was measured using Cronbach’s alpha to ensure that the items were internally consistent. The suggested value of Cronbach’s alpha that indicates satisfactory reliability must be at least 0.7 (George & Mallery 2003). The five criteria of the Likert scale were converted into numerical ranges: 4.21-5.00 for “totally agree”, 3.41-4.20 for “agree”, 2.61-3.40 for “neutral”, 1.81-2.60 for “disagree” and 1.00-1.80 for “totally disagree”. In addition, the questionnaire was designed as an internet-based study to minimise the conditions for social desirability bias; the internet-based questionnaire allowed the respondents to answer the questions anonymously without any social pressures. In addition, the questions were formulated in a neutral way to lower the respondents’ potential concerns about social judgement (Stuart & Grimes 2009, Krumpal 2013, Heerwig & McCabe 2009).

Data analysis

The data was analysed using IBM SPSS version 22 to test the three null hypotheses. Additionally, Cronbach’s alpha was used to examine the reliability of the questionnaire. The first hypothesis was that there would be no significant differences in pre- and post-mean ratings of students’ understanding of the flipped classroom. The second hypothesis was that there would be no significant differences in pre- and post-mean ratings of satisfaction with in-class activities in the flipped classroom. The final hypothesis was that there would be no significant difference in programming testing score measures between the cohorts of semester 1, 2016 and semester 1, 2017. A factor analysis identified the underlying dimensions of the questionnaire items. A one-way multivariate repeated measures analysis of variance tested the difference between the week 1 and week 15 questionnaire responses about students’ understanding of the flipped-classroom concept. One-sample t-tests tested whether students’ satisfaction with in-class activities provided in the flipped-classroom teaching method differed from the neutral rating. A one-way multivariate analysis of variance compared the examination and mini-test scores with the scores of students from the previous semester, who had learned in a conventional classroom.

Results

The number of students completing the online self-assessment pre-test and post-test

The number of times that students attempted to evaluate their understanding of the topic from their self-study was recorded on LMS2 (

Table ***1***).

Table 1. Number of attempts at the pre- and post-test as recorded by LMS2

Topic	No. of attempts	
	Pre-test	Post-test
Introduction to programming	150	119
Introduction to C programming	107	84
Data type, variables and expressions	190	116
Input and output	141	85
Selection control	101	70
Iteration control	100	74
One-dimensional arrays	116	76
Two-dimensional arrays	118	70
Pointer	229	116
Function	211	101
Structure	186	127
File	142	77

The reliability of the questionnaire

Cronbach's alpha was used to test the questionnaire's reliability. The alpha of the week 1 questionnaire was 0.70 (n=6), and that of the week 15 questionnaire was 0.97 (n=9). These values indicate the excellent reliability of both forms of the questionnaire.

Principle axes factor analysis

The items of the week 15 questionnaire were analysed using principle components factor analysis to identify the underlying dimensions of the 20 question items (Table 2).

Table 2. Descriptive statistic, communalities, and loading of items in the questionnaire

Item	Mean	Communalities	Factor 1	Factor 2
In-class activity created a fun environment	3.89	.857	.891	
In-class practice improved learning	3.94	.801	.864	
Self-evaluation used the post-test	3.92	.804	.852	
Using boxes helped in understanding the concept of arrays	3.89	.798	.852	
In-class activity helped with focus	3.92	.853	.841	
In-class activity created a link between theory and practice	3.87	.820	.821	
In-class coding improved learning	3.87	.713	.808	
Introductory brief helped in understanding the objectives	4.00	.712	.791	
Code debugging helped in understanding program operation	3.89	.774	.719	
Flipped classroom improved understanding of programming	3.83	.711	.692	
Diagram improved understanding of the program operation	3.87	.648	.631	
The level of familiarity used LMS2	4.02	.461	.604	
Pre-test helped to identify problems before class	3.77	.557	.570	
Pre-test helped with learning outcomes	3.81	.518	.513	

Item	Mean	Communalities	Factor 1	Factor 2
Learning material before class helped in understanding the topic	3.70	.604	.513	
The level of skill used IT	3.42	.860		.925
The level of skill coded a program	3.17	.792		.864
The level of understanding of flipped classroom concept used in learning process	3.57	.673		.675
Kahoot! improved attention during class	4.00	.634		.675
The level of skill used computer	3.83	.666		.655

As shown in Table 2, the communalities suggested that all question items related to the common themes of the questionnaire except for the item "The level of familiarity with LMS2" which showed only modest participation (communality = .46). This suggested that this item did not strongly contribute to measuring any common themes in the questionnaire. The three eigenvalues of the item called the level of understanding of flipped classroom concept, the level of computer expertise, and "The level of familiarity with LMS2" were greater than 1; two solutions were compared: one with two factors and one with three. The differences between the two solutions were slight, and conceptual simplicity suggested that the two-factor solution should be retained. In addition, the scree plot suggested a two-factor solution at most (Figure 1).

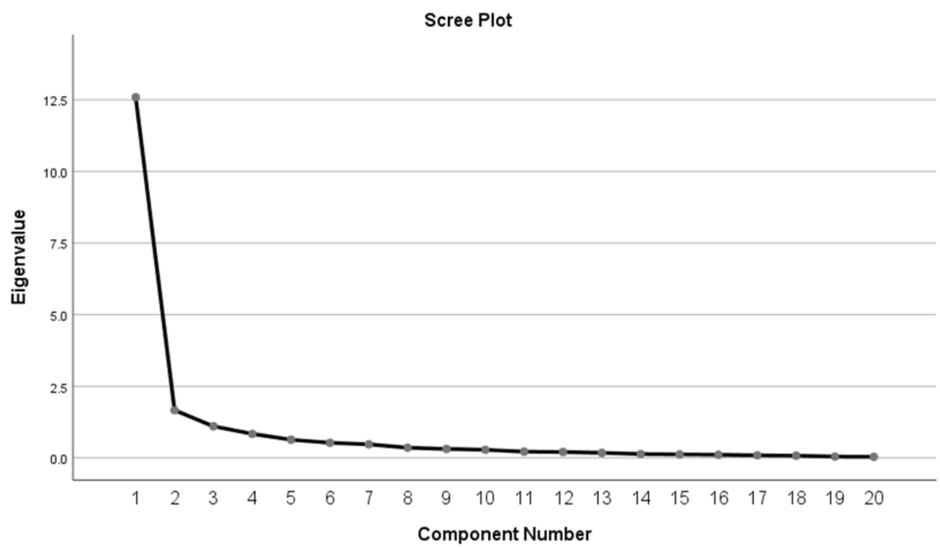

Figure 1. Scree plot of principle component factor analysis

The rotated factor matrix showed that, after Varimax rotation, Factor 1 accounted for approximately 45% of the questionnaire variance and Factor 2 for approximately 27% (

Table ***3***). In total, the two factors accounted for approximately 71% of the questionnaire variance.

Table 3. Rotated factor matrix

Component	Extraction sums of squared loading			Rotation sums of squared loadings		
	Total	% of variance	cumulative %	Total	% of variance	cumulative %
Factor 1	12.6	63.0	63.0	8.9	44.7	44.7
Factor 2	1.7	8.3	71.3	5.3	26.5	71.3

The plot of rotated factors in Figure 2 illustrates the variable loadings on the rotated factors. Factor 1 was characterised by high loadings of all "in-class" items, such as "In-class activity created a fun environment", "In-class practice improved learning", "In-class activity helped with focus", "In-class activity created link between theory and practice", and "In-class coding improved learning". As indicated by the post-test, students reported that using boxes helped them understanding the concept of arrays, and the introductory brief helped them understand learning objectives. Factor 2 was characterised by students' self-reported level of IT knowledge, programming knowledge, computer expertise and understanding of the flipped-classroom concept, as well as their opinion about whether Kahoot! improved attention. The variables did not particularly cluster or group, but were largely spread between the two factors, except that the level of IT knowledge and the level of programming knowledge clearly characterise Factor 2 as an independent grouping, while the "in-class" variables and self-evaluation using the post-test, whether using boxes helped in understanding the concept of arrays and whether the introductory brief helped in understanding the objectives clearly characterise Factor 1 as the other independent grouping. In summary, Factor 1 comprised participant ratings of how various activities helped students understand concepts, while Factor 2 independently comprised participant self-ratings of knowledge and expertise.

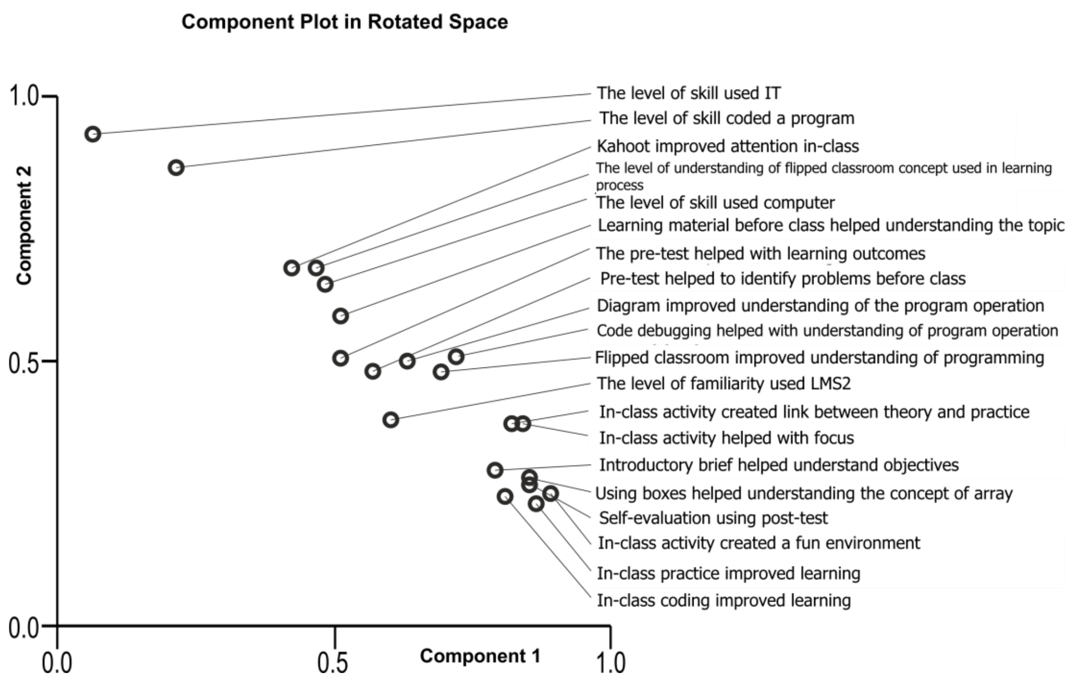

Figure 2. Component plot in rotated space

Students' self-rating report for understanding the flipped-classroom concept and IT knowledge

A one-way multivariate repeated measures analysis of variance tested the hypothesis that there was no difference in the mean self-ratings on the five items in the questionnaires administered in weeks 1 and 15 (The level of understanding of flipped classroom concept, The level of computer expertise, The level of familiarity with LMS2, The level of IT knowledge, and The level of programming knowledge). The descriptive statistics are shown in Table 4. The analysis showed that there was a statistically significant difference between the participants' week 1 and week 15 mean self-ratings across the five items, with Wilk's lambda = 0.03, $F(10,220) = 116.0$, $p < .001$. The results suggest that the self-rating scoring increased over the semester for each of the five items.

Table 4. Mean (standard deviation) of participants' self-ratings of knowledge and expertise

Item	Pre (*n*=63)	Post (*n*=53)	Total (*n*=116)
The level of understanding of flipped classroom concept used in learning process	3.14 (0.72)	3.57 (0.69)	3.34 (0.73)
The level of skill used computer	3.30 (0.68)	3.83 (0.91)	3.54 (0.83)
The level of familiarity used LMS2	3.63 (0.88)	4.02 (0.86)	3.81 (0.89)
The level of skill used IT	2.90 (0.82)	3.42 (0.89)	3.14 (0.88)
The level of skill coded a program	2.38 (0.87)	3.17 (0.85)	2.74 (0.94)

Students' satisfaction with in-class activities in the flipped classroom

A one-sample t-test was conducted to compare the mean opinion ratings and the neutral rating of 3 points. There was a significant difference in the mean opinion ratings and the neutral rating of 3 for all opinions (Table 5). These results suggest that all students' opinion ratings were higher than the neutral rating of 3. Specifically, the mean in the range of 3.41-4.20 indicated that students agreed that in-class activities help them to improve their understanding of programming and related concepts, improve learning outcomes and increase their engagement in class. In particular, the core elements with the highest mean score were the aspects of focusing in learning, as they had a clear understanding of the learning objective, and paying more attention in the class ($M = 4.00$), followed by the aspect of hands-on experience ($M = 3.94$). The aspect of self-evaluation also obtained a high mean ($M = 3.92$) indicated that the activities in class help students to improve their learning process.

Table 5. One-sample t-test of the mean opinion ratings in students' satisfaction with in-class activities

Item	Mean (SD)	*t*	*df*	*p*	Mean difference
Learning material before class helped understanding the topic	3.70 (0.93)	5.45	52	<.001	.70
The pre-test helped with learning outcomes	3.81 (0.73)	8.03	52	<.001	.81
The pre-test helped to identify problems before class	3.77 (0.87)	6.48	52	<.001	.77
The introductory brief helped in understanding objectives	4.00 (0.81)	9.00	52	<.001	1.00
Kahoot! improved attention in class	4.00 (1.04)	7.01	52	<.001	1.00
The diagram improved understanding of the program operation	3.87 (0.94)	6.71	52	<.001	.87

Item	Mean (SD)	*t*	*df*	*p*	Mean difference
Using boxes helped in understanding the concept of arrays	3.89 (0.93)	6.91	52	<.001	.89
Self-evaluation used the post-test	3.92 (0.98)	6.88	52	<.001	.92
Code debugging helped with understanding program operation	3.89 (0.89)	7.24	52	<.001	.89
In-class practice improved learning	3.94 (0.91)	7.57	52	<.001	.94
In-class coding improved learning	3.87 (0.92)	6.86	52	<.001	.87
In-class activities created a fun environment	3.89 (0.87)	7.42	52	<.001	.89
In-class activities helped with focus	3.92 (0.83)	8.12	52	<.001	.92
In-class activities created link between theory and practice	3.87 (0.86)	7.38	52	<.001	.87
The flipped classroom improved understanding of programming	3.83 (0.80)	7.53	52	<.001	.83

Students' examination score adapting the flipped classroom

To compare the examination scores between groups of students, an independent sample t-test was conducted to compare general academic ability as shown in the GPA (Grade Point Average in a 4.0 points GPA system; for example, 2.35) in three basic science subjects (mathematics, physics and chemistry) in the 2016 and 2017 cohorts. The analysis showed that there was no significant difference in the GPA of the 2016 and 2017 cohorts (M = .93, SD = .63 versus M = 1.12, SD = .54), $t(125) = -1.83$, p = .70. An independent samples t-test was also conducted to compare programming assessment scores. The results suggest that there was a difference between the 2016 and 2017 participants' programming assessment scores. The descriptive statistics are shown in Table 6. Specifically, the findings indicate that the average assessment score of the 2017 cohorts for mini-test was higher than of the 2016 cohorts, with an increase from less than 50% of the total average score of the mini-test to more than 50%. In addition, the average midterm score of the 2017 cohort was 11 units higher than that of the 2016 cohort. The average final score also increased in the 2017 cohort.

Table 6. Mean (standard deviation) of assessment score

Assessment	Cohort 2016 (*n*=74)	Cohort 2017 (*n*=64)	Total (*n*=138)
Mini-test (%)	42.1 (26.3)	57.6 (30.4)	49.31 (29.2)
Midterm (%)	36.3 (18.3)	47.3 (23.3)	41.4 (21.4)
Final exam (%)	31.3 (22.5)	38.9 (21.7)	34.8 (22.4)

Discussion and recommendations

This section discusses students' perceptions of the flipped classroom, their satisfaction with in-class activities and their learning achievement in the flipped-classroom environment. Overall, students had a positive perception of the flipped-classroom model. The findings revealed that students were satisfied with the flipped-classroom method, in which they studied the content at home and worked on the activities during class. In addition, students were satisfied with the in-class activities, with many agreeing that in-class practice, activities and exercises helped them to understand the topic and to apply the topic content to their coding. Moreover, external factors, such as a fun learning environment, that support the learning process helped students to improve their learning. The fun learning atmosphere was created by the lecturer's content-delivery approach, which included the voice and tone of explanation, the appropriate language for delivery content, the lecturer's appearance and activities such as game-based exercises. This is shown by students' comments in the open-ended questions in the questionnaire; for example, "The lecturer taught in an easy-going, lovely, and entertaining way, therefore, I understand the concept without stress", "I like the flipped classroom teaching and learning method", "I would like to do exercises in every session" and "I like flipped classroom because I can work and exchange ideas with my friends." These results are in accordance with the research of Kim et al. (2014), Puarungroj (2015), Campbell et.al (2014), Turan et. al. (2018) and Findlay-Thompson et. al. (2014), in which students were satisfied with the flipped-classroom activities because they provided an opportunity to gain exposure prior to class, an interactive approach for learning and support for building a learning community in which students practice or work together with their peers. Most comments were positive and consistent with the quantitative analysis, showing that the flipped-classroom approach increased students' motivation to learn and engage in the class and improved the learning environment.

With regard to students' learning achievements, the findings confirmed that students' scores were significantly higher when studying in a flipped classroom. This finding is consistent with the research of O'Flaherty and Phillips (2015), who reported that a flipped-classroom approach increased the academic performance of students as measured by improved examination results, pre-test and post-test scores or programming test scores. Despite the statistical significantly higher scores for the midterm and final programming test compared with the previous semester, one issue needs to be carefully considered. The low score for the final examination may result from the level of difficulty of the topics, which demanded a more logical understanding of the concepts and ability to apply them to various situations to do the coding. This research was undertaken because the students' assessment results were poor for the module concerned in the previous semester. With a conventional classroom in 2016, the overall assessment average percentage (mini-test, midterm, and final) was 36.6%. With a flipped classroom in 2017, the overall assessment average percentage was 47.9%. The statistical analysis showed that the student gain from 2016 to 2017 was highly significant, both overall and in each area of assessment: the first test, the mid-module test, and the module final exam. Therefore, the overall improved assessment average percentage is compelling evidence for the effectiveness of flipping the classroom.

It was interesting to find that the class schedule and material provided affected students' perception of the flipped classroom. Some students suggested that it would be more fun if the class were in the afternoon (the classes took place in the morning sessions). The reason may be that students felt that they were not ready to study in the early class because they did not feel fully awake. Some students suggested that more details should be provided in the PowerPoint slides.

The challenge in implementing the flipped-classroom approach in the course is ensuring that students prepare before they come to the class. The lecturer needs a systematic approach to keep students' engagement with the pre-class material. One of the approaches used in this study was giving credit points for completing the pre-test quiz before students came to class. However, this method might not work for a different group of students and might lead to a negative learning environment. Therefore, a positive attitude should be applied for implementing flipped classroom; for example, the process of implementing flipped classroom needs to be explained during the first session and systematic instruction needs to be provided to the students. This may help the students to have clear expectations from the course and a direction on how to learn in their own time. The results of this study suggest that the flipped classroom benefited the students in several ways: it improved their understanding of the concepts and increased their level of engagement in the class.

It is suggested to make the flipped classroom more effective for a programming class with several types of material provided to allow students to study ahead of class. In addition, if the class is in a morning session, the lecturer should employ more physical activities in the class, such as playing challenging games to find solutions, drawing diagrams or engaging in discussion.

Conclusion

In summary, this study provided an understanding of students' perceptions of flipped classrooms in an introductory course on structured programming. The results indicated that a flipped classroom holds significant promise for the successful improvement of students' learning output in programming courses. Students agreed that they understood the flipped classroom and were highly satisfied with the activities provided in class sessions. Students agreed that learning in a flipped classroom with in-class practice and out-of-class studying helped them to understand programming concepts and to implement programs from that understanding. In addition, after the students learned the course in a flipped-classroom environment, their programming skill as measured by their mini-test and programming examination scores was significantly higher than that of the students in the traditional-lecture class environment from the previous year.

References

Alhazbi, S 2016, 'Using flipped classroom approach to teach computer programming', *Teaching, Assessment, and Learning for Engineering (TALE), 2016 IEEE International Conference*, pp. 441-444.

Amresh, A, Carberry, AR & Femiani, J 2013, 'Evaluating the effectiveness of flipped classrooms for teaching CS1', *IEEE Frontiers in Education Conference*, pp. 733-735.

Beatty, B & Ulasewicz, C 2006, 'Faculty Perspectives on Moving from Blackboard to the Moodle Learning Management System', *TechTrends*, vol. 50, no. 4, pp. 36-45.

Bergmann, J & Aaron, S 2012, 'Flip your classroom: Reach every student in every class every day', *International Society for Technology in Education.*

Bishop, JL & Verleger, MA 2013, 'The flipped classroom: A survey of the research', *ASEE National Conference*, Atlanta, GA, pp. 1-18.

Bremer, D & Bryant, R 2005, 'A comparison of two learning management systems: Moodle vs Blackboard', *The 18th Annual Conference of the National Advisory Committee on Computing Qualifications*, pp. 135-139.

Butler, M & Morgan, M 2007, 'Learning challenges faced by novice programming students studying high level and low feedback concepts', *The 24th ASCILITE Conference*, pp. 2-5.

Campbell, J., Horton, D., Craig, M., & Gries, P. (2014). Evaluating an inverted CS1. *The 45th ACM Technical Symposium on Computer Science Education*, pp. 307-312.

Cole, J & Foster, H 2007, *Using Moodle: Teaching with the popular open source course management system.* O'Reilly Media, Inc., Newton, MA.

Crawford, JA & Kelder, J-A 2019, 'Do we measure leadership effectively? Articulating and evaluating scale development psychometrics for best practice', *Leadership Quarterly*, vol. 30, no. 1, pp. 133-144.

Dellos, R 2015, 'Kahoot! A digital game resource for learning', *International Journal of Instructional Technology and Distance Learning*, vol. 12, no. 4, pp. 49-52.

Dougiamas, M & Taylor, P 2003, 'Moodle: Using learning communities to create an open source course management system', *World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA)*, pp. 171-178, Chesapeake, VA.

Enfield, J 2013, 'Looking at the impact of the flipped classroom model of instruction on undergraduate multimedia students at CSUN', *TechTrends*, vol. 57, no. 6, pp. 14-27.

Ferreri, SP & O'Connor, SK 2013, 'Redesign of a large lecture course into a small-group learning course', *American Journal of Pharmaceutical Education*, vol. 77, no. 1, p. 13.
Findlay-Thompson, S & Mombourquette, P 2014, 'Evaluation of a flipped classroom in an undergraduate business course', *Business Education & Accreditation*, vol. 6, no. 1, pp. 63-71.
Fulton, K 2012, 'Upside down and inside out: Flip your classroom to improve student learning', *Learning & Leading with Technology*, vol. 39, no. 8, pp. 12-17.
George, D & Mallery, P 2003, *IBM SPSS statistics 23 step by step: A simple guide and reference,* Allyn & Bacon, Boston.
Gilboy, MB, Heinerichs, S & Pazzaglia, G 2015, 'Enhancing student engagement using the flipped classroom', *Journal of Nutrition Education and Behavior*, vol. 47, no. 1, pp. 109-114.
Graham, K 2015, 'TechMatters: Getting into Kahoot!(s): Exploring a game-based learning system to enhance student learning', *LOEX Quarterly*, vol. 42, no. 3, pp. 6-7.
Guo, PJ, Kim, J & Rubin, R 2014, 'How video production affects student engagement: An empirical study of MOOC videos', *First ACM Conference on Learning@ Scale*, pp. 41-50.
Hawks, SJ 2014, 'The flipped classroom: now or never?', *AANA Journal*, vol. 82, no. 4, pp. 264-269.
Heerwig, JA & McCabe, BJ 2009, 'Education and social desirability bias: The case of a black presidential candidate', *Social Science Quarterly*, vol. 90, no. 3, pp. 674-686.
Herreid, CF & Schiller, NA 2013, 'Case studies and the flipped classroom', *Journal of College Science Teaching*, vol. 42, no. 5, pp. 62-66.
Jonsson, H 2015, 'Using flipped classroom, peer discussion, and just-in-time teaching to increase learning in a programming course', *IEEE Frontiers in Education Conference (FIE)*, pp. 1-9.
Kim, M, Kim, S, Khera, O & Getman, J 2014, 'The experience of three flipped classrooms in an urban university: an exploration of design principles' *The Internet and Higher Education*, vol. 22, pp. 37-50.
Krumpal, I 2013, 'Determinants of social desirability bias in sensitive surveys: a literature review', *Quality & Quantity*, pp. 2025-2047.
Maher, M, Latulipe, C, Lipford, H & Rorrer, A 2015, 'Flipped classroom strategies for CS education', *The 46th ACM Technical Symposium on Computer Science Education*, pp. 218-223, Kansas City, MO.
O'Flaherty, J & Phillips, C 2015, 'The use of flipped classrooms in higher education: A scoping review', *The Internet and Higher Education*, vol. 25, pp. 85-95.
Prober, CG & Khan, S 2013, 'Medical education reimagined: a call to action', *Academic Medicine*, vol. 88, no. 10, pp. 1407-1410.
Puarungroj, W 2015, 'Inverting a computer programming class with the flipped classroom', *Proceedings of the International Conference on eLearning for Knowledge-Based Society*, pp. 40.1-40.7.
Rendersoft Development 2019, *CamStudio Open Source*, viewed at https://camstudio.org/.
Rovinelli, RJ & Hambleton, RK 1977, 'On the use of content specialists in the assessment of criterion-referenced test item validity', *Dutch Journal of Educational Research*, no. 2, pp. 49-60.
Stuart, GS & Grimes, DA 2009, 'Social desirability bias in family planning studies: a neglected problem', *Contraception*, vol. 80, no. 2, pp. 108-112.
Talbert, R 2017, *Flipped learning: A guide for higher education faculty.* Stylus Publishing, LLC, Sterling, VA.
The Code::Blocks Team 2019, *Code::Blocks [version 16.01]*, viewed at http://www.codeblocks.org/home
Tune, JD, Sturek, M & Basile, DP 2013, 'Flipped classroom model improves graduate student performance in cardiovascular, respiratory, and renal physiology', *Advances in physiology education*, vol. 37, no. 4, pp. 316-320.

Turan, Z & Göktaş, Y 2018, 'Innovative Redesign of Teacher Education ICT Courses: How Flipped Classrooms Impact Motivations?', *Journal of Education and Future*, vol. 13, pp. 133-144.