International Journal on Advanced Science Engineering Information Technology

Vol.12 (2022) No. 2
ISSN: 2088-5334

A Flipped Classroom Framework for Teaching and Learning of Programming

Rosnizam Eusoff[a,*], Abdullah Mohd Zin[b], Syahanim Mohd Salleh[b]

[a] Center for Software Technology and Management, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia
[b] Universiti Kebangsaan Malaysia, Bangi, 43000, Malaysia
Corresponding author: [*]zameusoff@gmail.com

Abstract—**Programming is a difficult subject that requires high analytical skills to comprehend students. Various strategies are introduced to improve difficulties in learning programming for students. This study examined four flipped classroom frameworks and was triangulated with the findings of document analysis from 44 articles related to flipped classrooms in programming. The articles were coded using five different codes emergent from the research questions. The coding produced 314 quotations related to the research questions. The purposes of the flipped classroom are to prepare students before class, give extra in-class time, and create an active learning environment in the classroom. The flipped classroom was implemented mostly in two phases: pre-class sessions and in-class sessions, with individual and collaborative work activities. Ensuring the participation of students in the activities outside the classroom is the biggest challenge in a flipped classroom, while preparing tools and activities is cumbersome to some lecturers. The pre-class activities focus on the introduction and theoretical topic. The implementation of flipped classrooms reported improving students' skills, competencies, and satisfaction in programming. The potential of exploring analysis and problem-solving activities in pre-class sessions while strengthening students' knowledge in post-class sessions seems to be assured. A flipped classroom framework for programming is proposed. The framework has three phases; pre-class, in-class, and post-class, with suggestions, activities, and motivational elements to complement the conceptual and technical needs in programming using flipped classroom approach.**

Keywords—**Learning of programming; flipped classroom; active learning; collaborative learning.**

Manuscript received 9 Apr. 2021; revised 30 Jun. 2021; accepted 18 Nov. 2021. Date of publication 30 Apr. 2022.

I. INTRODUCTION

Programming is a systematic and scientific process that requires high analytical skills. Programming is also considered a challenging subject for students and teachers [1]. Programming is also considered one of the biggest challenges in computing [2]. Various studies concluded that programming is difficult to learn [3]–[9]. The difficulty in mastering these programming skills stems from several problems that previous researchers have identified. In solving this problem, various strategies are used in teaching and learning programming [10]–[14]. In general, the challenges faced by students in programming are classified into three main categories: syntactic knowledge, conceptual knowledge, and strategic knowledge [15], [16]. Most of the problems encountered are related to the level of existing knowledge possessed by the students [16]. In contrast to traditional teacher-centered learning methods, the flipped classroom approach is a method used to enhance students 'learning experience by strengthening students' self-learning outside the classroom [17]–[19] and active learning while in the classroom [20]–[22]. Active learning is one of the strategies that can improve students' understanding of programming [23].

There is an increasing interest in using the flipped classroom (FC) approach in teaching among educators [24]. This situation is supported by advances in communication information technology and the need for change in teaching and learning. A few flipped classroom frameworks and models are used as general guidelines in implementing FC in teaching and learning [25]–[29]. However, there are still shortcomings in terms of the implementation framework, theoretical limitations, and empirical evidence of the effectiveness of FC in teaching and learning [19], [29]–[31].

The study on the implementation of flipped classrooms for programming was conducted through document analysis of 44 systematically selected research papers. The analysis findings are used to answer some research questions that have been constructed for the implementation of the study. A flipped classroom framework for programming is proposed based on the analysis findings. While identifying the suitable frameworks for teaching and learning programming, four FC frameworks and models were examined [25]–[27], [29]. The study aims to provide a workable framework for the FC approach in programming.

The use of FC was recorded in 1995 when a lecturer began inserting his subject notes into a university computer network. It is intended that the notes required to be read in advance by the students before face-to-face meetings in class [32]. Subsequently, the flip approach expands and has been widely used in various fields [33], [34]. The term *'flipped classroom'* was coined by Bergmann and Sams, two high school teachers, in 2012 [30]. Active learning has an active student involvement in the learning process that takes place [35]. Active learning also increases student involvement in the learning environment and improves the learning process as well as outcomes from learning [36]. FC provides an opportunity to create an active learning environment centered on students outside the classroom as well as in the classroom [37]–[40]. FC also enhances the programming experience through active involvement in learning [18]. Through FC part of the learning process will take place outside the classroom. While learning in the classroom can be focused on more effective learning activities such as correcting misunderstandings about concepts as well as allowing students to organize their learning [21], [22], [42]. In general, FC is perceived to positively affect students' knowledge, skills, and involvement in learning [25]. FC frameworks and models have been used for various subjects and courses.

A. Murillo-Zamorano et al. [25] *Theoretical Framework*

Murillo-Zamorano *et al.* [25] have outlined the impact of FC on students' knowledge, skills, and engagement as in Fig. 1. FC directly impacts students' knowledge through active learning involves. FC enhances students' ability to learn on their own and, at the same time, increases their collaborative ability in groups [31]. FC also has a positive impact on students' skills. Student engagement and accountability in learning are higher than traditional methods [42].

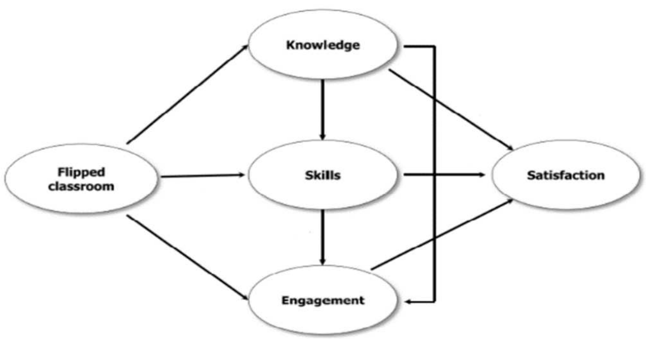

Fig. 1 Murillo-Zamorano *et al.* [25] Theoretical Framework

At the same time, FC allows students to apply their skills by using learning materials such as videos under their control. This situation allows the students to determine the learning process outside the classroom independently. Skills developed by students directly have a positive relationship to students' learning engagement. This positive relationship occurs due to increased understanding of the lesson content and the relationship through group collaboration. The positive relationship between knowledge, skills, and engagement gained through FC is directly related to students' learning satisfaction. Fig. 1 is a theoretical framework developed by Strayer [26], showing the relationship between FC and related constructs in learning.

B. Strayer's Flipped Classroom Conceptual Framework

Strayer's flipped classroom conceptual framework is one of the earliest frameworks introduced for a flipped-classroom approach in teaching and learning. The rapid growth of technological developments creates opportunities to turn one-way learning methods into a more interactive learning environment. The use of educational technology in delivering learning content outside the classroom is the main idea that FC incorporates in teaching and learning. Through FC, students could experience the learning process with the activities that are carried out by using technology. The use of FC is made based on the idea to provide opportunities for the active involvement of students in the classroom. This framework lays two main constructs in the learning process of using FC, namely activities and the use of technology. In turn, this situation will affect the elements in the learning environment. Fig. 2 is a theoretical framework to describe the processes involved in learning through FC.

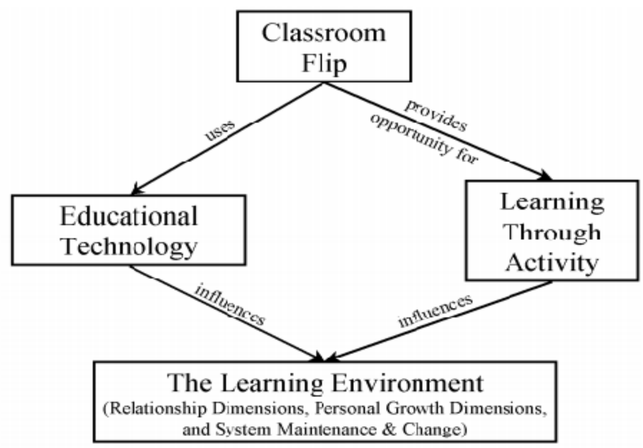

Fig. 2 Strayer's flipped classroom conceptual framework [26]

C. Strayer's Theoretical Framework for The Universal Principles for Flipped Instruction

The theoretical framework for the universal principles for flipped instruction was introduced by Strayer [26] in 2017. The interaction between the delivery of information outside the classroom and the experience gained while in the classroom are the benefits gained. FC enables students to acquire early knowledge and strengthen their understanding before and after class outside of the classroom. Active learning is emphasized where students must take responsibility for their learning outside or inside the classroom. However, teachers need to prepare the students to carry out the learning activities outside the classroom by giving effective instructions while in the classroom. The basic principle of FC is to increase students' interaction actively between students and teachers. Collaborative activities in the classroom with peers for problem-solving are highly emphasized. Activities outside the classroom can encourage reflection and reaction from students to learning in the classroom. Activities can also be designed to strengthen students' understanding of concepts they have learned while in the classroom. Classroom assignments are designed to build new knowledge as part of the learning community with classmates. Fig. 3 is a theoretical framework for the universal principles for flipped instruction.

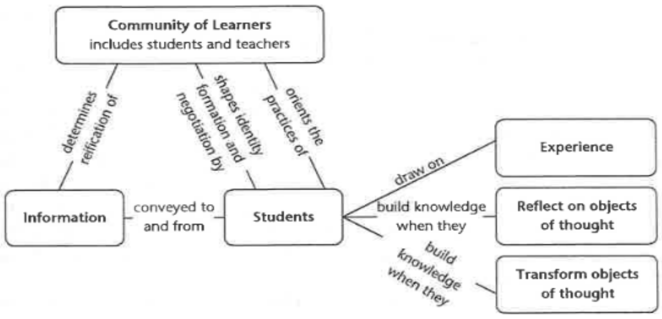

Fig. 3 The Theoretical Framework for The Universal Principles for Flipped Instruction [26]

D. Lo & Hew A Flipped Classroom Approach Model

Lo & Hew proposed two main parts of FC, namely extracurricular and classroom learning as shown in Figure 4. Outside classroom activities are focused on direct instruction. The activities are carried out at the level of remembering and understanding in learning taxonomy. Apart from watching videos, activities such as reading on related topics in books and articles, online exercises, and quizzes can also be done outside the classroom [43], [44].

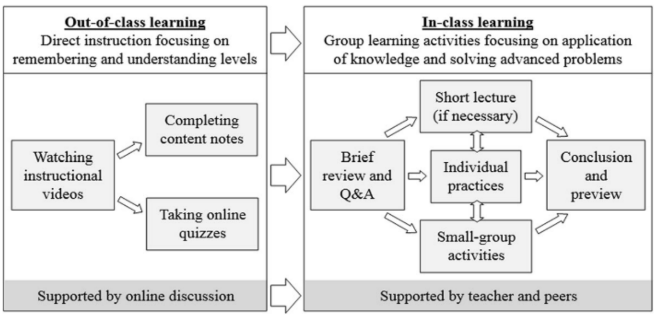

Fig. 4 Lo and Hew A Flipped Classroom Approach Model [29]

Online discussions will support the activities that have been carried out. As for the classroom activities, group learning activities will be focused on the application of knowledge as well as more challenging problem-solving involved. Among the activities are brief reviews, question and answer sessions, individual exercises, group activities, and summaries. Teachers and peers will assist with these activities. The basic model of FC is developed, focusing on the primary group of students (K12). Fig. 4 is the proposed flipped classroom approach model.

II. Materials and Method

A. Purpose of The Study

The study aims to propose the FC framework for the teaching and learning of programming. This framework is intended to be used as a guideline in developing activities and materials for teaching and learning programming. Below are the research questions (RQ) used in the study.

RQ1.What is the purpose of using flipped classrooms in programming?
RQ2. How is flipped classroom implemented in programming?
RQ3. What are the activities of the flipped classroom in programing?
RQ4. What are the challenges and effects of the flipped classroom in programming?

B. Document Analysis

Document analysis is a method used in analyzing data from selected articles in the study. There are 44 articles related to the implementation of FC in the teaching and learning of programming, which has been selected systematically. Document analysis is a replicable systemic research technique and draws valid conclusions based on text or non-text material in an appropriate context. Document analysis should also be guided by clear research questions [47], [48]. Different codes are used to facilitate the process of classifying quotations accordingly. For the first research question (RQ1), the purpose of FC in programming, the code Purpose is used. The highlighted quotations about the purpose of using FC found in the articles were put under this code. For the second research question (RQ2), code Implementation is used to gather relevant quotations on FC implementation in programming. For the third research question (RQ3), the activities involved in implementing FC code Activity are used. For the fourth research question (RQ4), the challenges and effects of FC in programming, code Challenge, and code Effect are used. All 44 articles were analyzed, and the relevant quotations in the articles were placed following the suitable code using ATLAS.ti 8.

C. How Articles were Selected for the Research

The selection of related articles to the study was adapted through the guidelines used by Preferred Reporting Items for Systematic Reviews and Meta-Analyzes (PRISMA), which involved four processes, namely identification, screening, eligibility, and include. Searching for the document using this method can produce articles related to the study. It also helps to identify the criteria of articles accepted (inclusion) or rejected (exclusion) in the search [47]. In the first stage of article searching, three main electronic databases are used: Scopus, Web of Science, and IEEE Xplore Digital Library. These three databases also have an advanced search function that helps search for articles specifically needed. In the next stage, ACM Digital Library is used with Google Scholar to review the articles that have been obtained through the specified databases. Searches are limited to articles published from 2010 to 2020. It corresponds to a maximum period of 5 to 10 years for articles used in systematic reviews [48]. The selection of articles is made based on the type of article that is from the journal or proceedings only. The selected articles are also those that are published in English only. At the same time, the selected articles are limited to the implementation of FC in teaching and learning programming. Selected articles are from the publication of various proceedings and journals. However, no selected articles were published in 2010 and

2012. There are 18 articles from the journal and 26 from the proceedings selected for document analysis in the study.

D. Data Coding and Analysis Process

The use of computer-aided qualitative data analysis software (CAQDAS) helps classify, organize, and analyze the data effectively, accurately, and more comprehensively in the study [49]. Data from 44 articles were analyzed by using ATLAS.ti 8 to facilitate the management of large amounts of data. As for data validation, three quotations from each code are randomly selected and given to two experts (rater 1 and rater 2) for verification of citation selection. SPSS 20 software is used to determine the inter-rater reliability of the data. The Kappa value generated through the analysis is 0.7 for the 5 codes used to classify the quotations. Kappa statistical values between 0.61 and 0.80 are accepted as substantial values, while values 0.81 to 1.00 are accepted as almost perfect [50]. Based on the codes used to analyze the selected articles, 314 related citations have been identified from 44 articles. Activity code has the highest number of quotations which are 36% or 113 quotations. Meanwhile, the Implementation code has 34% or 108 quotations with the second-highest quotations. Purpose code is 11% or 35 quotations, Challenge code is 10%, and Effect code is 9%, representing 30 and 28 quotations, respectively.

III. Results and Discussion

Based on the quotations that have been categorized according to code, further discussion is made according to the constructed research questions. The following are the results and discussions according to the sequence of research questions.

A. The Purpose of Using FC in Programming

The introduction of a new strategy is usually to improve the learning process or overcome difficulties in the previous approach. The implementation of FC as a strategy also has a similar reason. Quotations related to the purpose of FC are identified in the selected articles. The quotations are placed under the Purpose code. There are 35 quotations related to the purpose, as in Table 1. The purpose of FC in programming can be divided into two sub-categories, namely Pedagogy and Others. Through FC, most of the time in the classroom is used for activities that develop active learning amongst students [51].

TABLE I
Purpose of Flipped Classroom in Programming

Categories	**Sub-categories**	***f***	**%**
Pedagogy	Active learning	9	20.43
	Student-centered learning	6	13.62
	Blended learning	4	9.08
	Collaborative learning	2	4.54
	Positive learning environment	1	2.27
Others	A new learning approach	7	15.89
	Extra time in the teaching and learning process	5	11.35
	Improving students' self-engagement in teaching and learning	1	2.27

Through FC as well, passive learning content is removed from classroom sessions and provides extra time for active learning activities [52]. In the pedagogical category, active learning is the main purpose of FC in teaching and learning programming, which is 20.43%. According to Yan and Cheng [51], there is a pedagogical change from a teacher-centered learning approach to student-centered learning. Students need to perform pre-class activities such as watching videos or getting information online to gain basic knowledge. Students can watch or repeat the video at their discretion [18]. Meanwhile, face-to-face classroom sessions are used to carry out activities that help the student strengthen and master the knowledge [53]. The emphasis on the student-centered learning process in FC has a second-highest purpose of 13.62%.

In FC, students will receive early exposure before class by watching videos or accomplishing other activities related to their learning. Face-to-face classroom sessions are dedicated to problem-solving activities, feedback sessions, and discussions [54]. The blended learning offered in FC is the third-highest purpose FC is chosen which is 9.08%. Collaborative learning activities that can be done through FC are also the purpose of the approach used in teaching and learning programming, which is 4.54%. Collaboration between students occurs in learning activities in the classroom [55], such as paired programming in a computer lab [56]. A better learning environment and social interactions in paired learning can increase students’ achievement, confidence, and interest in programming [57]. A positive learning environment exists when students have enough time to conduct questions and answer sessions with lecturers and discussion sessions with peers in the classroom [58]. During a face-to-face session, there is more time because some of the learning content has been covered before the class. FC is a popular new learning approach that transforms traditional learning methods into student-centered active learning approaches [59]. As a new learning approach, it is also the purpose of implementing FC in Teaching and learning programming, which is 15.89%. A busy curriculum with a variety of subjects makes it difficult for extra time to be allocated for programming [60].

In the FC approach, part of the learning process is conducted outside the classroom. This situation will provide extra time in the classroom for learning through a variety of more focused activities [53]. The increase in time that can be obtained is also the purpose of introducing FC in teaching and learning programming, which is 11.35%. The FC model allows students to apply high cognitive skills in the classroom through the teacher’s guidance. Students can monitor their learning development based on the competencies that need to be achieved in the process [54]. Self-involvement or self-engagement is the purpose of FC in one study or 2.27%. In general, the main purpose of selecting FC as a strategy in teaching and learning programming is to create student-centered active learning. Active learning is a new learning method used to increase students’ involvement in the learning process itself. Time constraints and lesson content that need to be conveyed in the classroom are also problems FC can overcome as part of the learning process is conducted outside of class time. The extra time gained in the classroom can be used to carry out active learning activities and reinforcement of the lesson. Learning activities in FC can be implemented before class, during class, and after class. However, the results of the analysis found that most of the activities through FC are carried out before class and during class. Discussions about after-class activities are not much discussed based on the results of the analysis.

B. Implementation of FC in Programming

Implementation is an important process in the newly introduced approach. Strategic and planned implementation can help the success of the newly introduced approach. One hundred eight related quotations have been identified in the excerpts placed under the code of Implementation as in Table 2.

TABLE II
IMPLEMENTATION OF FLIPPED CLASSROOM

Categories	***f***	**%**
Video	42	95.34
Online quiz	26	59.02
Presentation slides	5	11.35
Notes	4	9.08
Learning portal	3	6.81
Worksheets	3	6.81
E-book	3	6.81
Credits / extra marks for participation	3	6.81
Interesting learning environment	3	6.81
Collaborative activity	3	6.81
Animation / interactivity	2	4.54
Systematic explanations to students about activities	2	4.54
Video screening period/activities	2	4.54
Appropriate activities	2	4.54
Activities/quiz after watching video	2	4.54
System log-in records	2	4.54
Automate tasks reminder	1	2.27

Various tools such as videos, presentation slides, quizzes, digital documents through various online learning platforms have been used when implementing FC. In the FC learning environment, students conduct self-learning outside the classroom through video viewing and subsequently achieve the student-centered active learning environment in the classroom [61]. The use of video in FC provides various advantages in teaching and learning that cannot be obtained through conventional methods [62]. Learning content can be viewed before class according to the time and appropriateness of students' level of understanding [53]. The videos are produced from personal recordings [63] as well as from the internet and various sources [44], [58]. The suitability and convenience obtained in the video have made it the most widely used tool in FC, which is 95.34%. In traditional learning methods, quizzes are used to test students' comprehension quickly. Quizzes are also used as a method of control as well as motivation for students to maintain their focus when learning. Answers from quizzes are used to measure students' understanding of new knowledge. Online quizzes are used to test comprehension and also as a method to ensure that students watch the video outside the classroom [53]. There are 59.02% uses of quizzes as material in FC for programming. Currently, presentation slides are one of the teaching aids that have been widely used in teaching and learning. The use of presentation slides as material to be read by students before class is 11.34%, while the use of learning notes as material in learning before class is 9.08%. The use of tutorial worksheets in the form of assignments that need to be completed before class, reading from e-books, and interactivity elements used as learning material have the same percentage of 6.81%.

There are two types of motivation in learning: intrinsic and extrinsic. Intrinsic motivation occurs in the satisfaction gained from the learning process without external causes. On the other hand, extrinsic motivation is the impulse of external factors in performing activities to obtain a result [64]. Motivation to encourage students' involvement in FC was identified in selected articles. Motivation is important and directly bears student learning attitudes and behaviors. Motivation is also an important cognitive factor in determining success in learning and positively impacts programming learning [15]. FC motivates students to learn on their own outside of the classroom and actively engage in in-class activities. The learning environment created by FC helps to enhance students' understanding. Giving credit points to students who complete the pre-quiz before class can increase some students' motivation [15]. At the same time, the marks are given in self-training before class also increase students' motivation in FC activities [65], [66]. To attract students to watch the video at home, learning objectives for the next class are given to the students before the class ends. Despite feeling a little stressed in the new learning style, the students thought they could learn faster and easier through FC [67]. This situation can also motivate students' involvement outside the classroom. During classroom learning sessions, students are actively involved in the inquiry and problem-solving activities, constructing their knowledge, working with peers, and reflecting on the learning process [68]. Collaborative activities between students are encouraged while motivating students to be actively involved in learning [69]. Students who have completed their assignments are encouraged to help friends who have not yet completed their assignments to encourage positive interaction and communication [65].

Some control strategies have been reported in several analyzed articles to ensure that students will perform planned activities outside of the classroom. Several control methods are used to ensure that students will be engaged in activities outside the classroom. The control methods used are also identified in these articles. As stated earlier, video is the most widely used tool in FC. Several strategies are used as controls in the use of video as the tool in implementing FC. Long video viewing periods will distract the viewer. To overcome this issue, video is limited to not more than 15 minutes so that students do not lose focus [60]. Automatic reminder methods are also used to remind students to watch videos before attending lectures [18]. Students need to make notes that will be submitted to the lecturer based on the video they have watched, and answering a short quiz after watching a video is also a form of controlling method [70]. There are also control methods that are embedded in the platform or portal that is used for learning. Students who are logged in to the system will be recorded as their attendance. A specific period is also set for uploading assignments through the system as a control to ensure students log in to the system [71], [72].

Video is the most widely used tool; however, the usage of the video should be appropriate in terms of the duration of the content and relevance to the learning topics [43]. Based on the study, the duration of the video screening should not exceed 15 minutes for each screening so that students can give full focus to this view in line with the period recommended by [58]. Furthermore, to ensure students' involvement in activities, especially outside the classroom, the lecturer used elements of motivation and control. In conclusion, the implementation of FC can be categorized under three categories, namely tools (materials), motivation, and control. However, the analysis of the documents that have been made found that the description of motivation and control used in FC implementation is limited to a few studies only.

C. The Activities of FC in Programing

Various activities before, during, and after class are conducted in the flipped classroom approach. Pre-class activities mean self-study done by students outside the classroom. In-class activities refer to face-to-face activities that take place in the classroom involving students and lecturers. In comparison, the post-class activity is a self-learning activity performed by students after class. Table 3 contains a list of FC activities in teaching and learning programming carried out outside the classroom and in the classroom.

TABLE III
ACTIVITIES IN FLIPPED CLASSROOM APPROACH

Phase	Activities	*f*	%
Pre-class	Watching video	37	83.99
	Online quizzes	15	34.05
	Reading (notes, slides, books, etc.)	9	20.43
	Completing tasks	5	11.35
	Online discussions	1	2.27
In-class	Active learning / collaborative learning	20	45.4
	Pair programming	9	20.43
	Quiz (online / offline)	8	18.16
	Discussions in group	3	6.81
Post-class	Revision on learning material	3	6.81
	Peer discussion*)*	2	4.54
	Exercises	1	2.27

In general, the teaching and learning of programming through conventional methods involves lectures, programming assignments, and written exams. FC transforms the one-way approach to group learning into a directed individualized learning method outside and inside the classroom. Furthermore, students will be involved in dynamic and creative activities that support an active learning environment, such as peer-assisted learning, cooperative learning, problem-based learning, group discussions, and group problem-solving [58]. Watching videos was the most reported activity in the study in pre-class activities at 83.99%. The second highest activity was the online quiz at 34.05%, and the activity of reading learning materials was the third highest at 20.43%. Active and collaborative learning is the most reported activity for in-class activities at 45.4 %, followed by online or offline quizzes at 20.43 %. For post-class activities, revision on learning material has the highest number of reports from the study, which is 6.81%, followed by discussions with friends of 4.54% and reinforcement exercises of 2.27%. In general, the post-class phase is not much discussed in most studies reviewed.

Alhazbi [15] reports some activities made during the implementation of FC in learning. Students are divided into several groups to complete a small programming project. Students are also given case studies, and discussions in an active learning environment between students are created. Each student has a learning journal that is used to record their learning progress. Formative assessments of students' comprehension are made weekly to identify students' performance. The effectiveness of implementing FC in learning will depend on the activities that are being carried out while in the classroom [73]. In contrast to traditional teacher-centered learning methods, activities in FC emphasize student-centered active learning and collaborative learning. While in the classroom, lecturers need to increase physical activity such as challenging games to solve problems, draw diagrams, or create discussion sessions [58]. According to Mohamed [74], pair programming is also good for encoding activities. Weak students will be paired with skillful programming students when performing coding tasks. Students will switch roles as drivers or observers while doing the coding tasks [74], [56]. Students will discuss exchange views and ideas to complete the assignments. Typically students do not encounter problems with learning materials until they are asked to apply the new knowledge that they have acquired to solve the problems [67].

D. The Challenges and Effects of FC in Programming

Challenges are the issues faced in implementing new approaches and strategies in teaching and learning. The implementation of FC in teaching and learning programming also faces some challenges and limitations. The effects of FC implementation in teaching and learning are also identified through reports from the analyzed articles. Table 4 contains a list of challenges faced and the effects of FC implementation in teaching and learning programming.

TABLE IV
CHALLENGES AND EFFECTS OF FLIPPED CLASSROOM

Categories	Sub-categories	*f*	%
Challenges	Challenge to ensure student involvement in pre-class and post-class activities	9	20.43
	An additional burden for the preparation of teaching and learning materials	5	11.35
	Students do not understand FC learning methods	4	9.08
	Problems from the materials used i.e., video quality, time, material limitations	4	9.08
	Difficulties in planning active learning activities	3	6.81
	Lack of communication skills between teachers-students / students	2	4.54

Categories	Sub-categories	*f*	%
	Higher cost to implement FC	1	2.27
	Lack of infrastructure	1	2.27
	Lack of skills in using technology in teacher/student	1	2.27
Effects	Improve the quality/environment of programming learning	8	18.16
	Improve test/examination results	7	15.89
	Positive impact/motivation to students in programming	7	15.89
	Improve students' understanding and self-engagement	6	11.35

The main challenge in FC is to ensure that students complete pre-class assignments and activities before entering the classroom which reported as much as 20.43% [74], [58], [75], [76], [77], [78]. It is in line with the biggest challenge of using the FC approach reported by [79] in their study. While the process of preparing new course content requires much effort, learning materials also need to be constantly updated due to the rapid development of programming languages [18], [65], [43]. Planning active learning activities in the classroom also gives the lecturer additional challenges and extra workload. This is reported as the second-highest challenge in FC, which is 11.35%. Lecturers or instructors need to have a systematic approach to ensure students' involvement in pre-class preparation [74], [58], [69]. Video is the most widely used tool in FC for teaching and learning. However, one of the disadvantages of watching videos in FC is that this activity will be done alone [44]. The challenge is to make sure students watch the video before class. According to [66], instructions to students to watch videos outside the classroom without a clear explanation resulted in students not doing those instructions. Lecturers need to be creative and innovative in planning the learning activities, especially for activities conducted outside the classroom. Challenge from students' understanding of FC learning methods and problems from learning materials have the same number of reports of 9.08%. Another challenge reported were communication problems between lecturers and students [80] which is 4.54%. Lack of skills in conducting online learning by lecturers and students. Lack of IT infrastructure to support the implementation of FC and the quality of learning materials as well as FC learning model is not yet well established [81].

There are various limitations in the implementation of FC in teaching and learning programming that can be identified. For active learning activities involving many students, the assistant lecturer may be needed to ensure effective learning happens in the classroom [74]. Some students are unaware that learning content and activities that need to be done outside of the classroom are also part of the learning process. This situation indicates the need to explain flip pedagogy clearly to students [65]. Usually, students who are unable to complete their assignments in the classroom do not prepare outside the classroom prior to attending the class [76]. Some students still cannot adapt to the FC method and think that the learning process is only through reading and completing the assignments given by the lecturer. Weaknesses in interpersonal skills among students as well as difficulties in communicating in discussions or writing in online forums, are also some of the limitations [82]. At the same time, feedback from students about the FC method used should also be considered by the lecturer. However, the feedback is not considered as the main source due to the limited understanding amongst students about the content and pedagogy of FC [43]. Without appropriate support tools, lecturers need to take a long time to provide course content in implementing FC. Appropriate support tools need to be developed to help lecturers to overcome such limitations in the future [71]. The quality and learning environment of programming were reported to improve using FC (18.16%). The use of FC improves examination results and positively impacts and motivates students in programming. Both have the same reporting percentage of 15.89%. FC also improves student's understanding and self-engagement which has a reporting percentage of 11.35% [74], [58], [67], [70]. FC is also found to improve students' collaborative skills and computational thinking [78]. At the same time, the initial preparations made by the students and the discussions in the classroom enhance meaningful learning that takes place in the classroom [18]. Activities outside and inside the classroom help students to understand the concept of programming and implement the concept during coding. In-class interaction between lecturers and students through FC activities increases the rate of students' involvement and interest in learning [83]. FC enhances understanding of programming concepts and increases students' engagement which has contributed to a positive impact on students' achievement [84]. The FC model also effectively enhances students' engagement, interaction, self-efficacy and attitudes. To all relevant elements are the main components of instructional design which is the key to the success of instruction in learning [78].

The study also reported increasing students' knowledge, in theory, coding skills, and competencies [60], [54]. Compared to traditional and FC methods, compared to students who implemented FC [53]. The implementation of FC has influenced students' attitudes and confidence towards programming [51]. In FC also, students can individually review topics that they had learned according to the suitability of their own time, while lecturers may benefit from the time in the classroom to focus more on real problems in learning [43]. The use of FC strategies positively affects students' attitudes and learning satisfaction. This condition has increased students' motivation and performance in programming. Watching a video about a related topic can improve skills and concepts in programming before attending classes [44]. Overall there is a change in students' achievement who received FC exposure as compared to traditional methods [85]. Analysis of the study shows that FC is suitable to be used and one of the potential strategies in teaching and learning programming [86]. The results found that FC can improve students' programming skills and coding comprehension and help them learn more effectively with better learning achievement [71].

E. Discussion on Findings

The development of a systematic learning environment design is important to optimize learning activities outside the classroom as well as in the classroom. Activities outside the classroom can take place pre-class or post-class. In comparison, activities in the classroom are face-to-face meetings between lecturers and students. Previously, based on the reports on the implementation of FC in programming, learning content outside the classroom is more focused on theory and a basic identification of sub-topics in programming. Face-to-face sessions in the classroom are focused on more difficult topics such as problem-solving analysis and program coding. Out-of-class learning content for sub-topics involving higher cognitive skills in programming can be enhanced according to the appropriateness of learning. Reinforcement and review activities made after class can enhance the learning content. Tools are important in implementing FC in learning, especially for activities outside the classroom. Video is the most widely used material in FC. However, the duration of the video should be appropriate to students' ability to focus on self-learning. An effective video screening duration is suggested not to exceed 15 minutes for a single screening period. As a control method, simple quizzes can be placed between video showtimes or after the show. Other learning materials used should also be appropriate to the learning objectives. The development of materials should be done according to the instructional design to provide an effective and efficient learning process. Activities for learning before, during, and after class are developed according to the learning objectives and the sequence of complexity from easy to difficult [28].

The main challenge that has been identified in the implementation of FC is the challenge to ensure students' preparedness before class and students' ability to carry out activities outside the classroom without the presence of teachers. Elements of motivation, either in the form of intrinsic or extrinsic motivation and control strategies, can be used to overcome the challenges in FC. These elements of motivation and control can be incorporated through the materials and activities carried out. FC learning emphasizes active learning in the classroom through a variety of collaborative activities in pairs and groups. Pair programming, group discussion, and problem-solving can be done while in the classroom. FC also emphasizes the active involvement of students in their learning process. The roles of teachers are to build scaffolds and strengthen students' understanding of the subject through discussion and criticism in learning activities in the classroom. At the same time, pre-class activities are built-in preparation before class, while post-class activities are built to strengthen students' understanding. Active learning is fundamental or core to the implementation of FC in teaching and learning. Active learning involves various collaborative activities, which is also an advantage in FC. In constructivist theory, understanding and knowledge are built through experience and reflection. Self-learning outside the classroom and active learning through FC can improve students' understanding and knowledge of programming.

Meanwhile, classroom sessions are focused on the level of knowledge and skills. Classroom activities focus on problem-solving skills as well as program coding. Students carry out collaborative activities in pairs or groups according to the suitability of the topics discussed. A study of the approach used in programming subject found that co-operative learning has the highest impact on improving achievement [13]. This situation indicates the need to create more co-operative learning activities during face-to-face sessions in the classroom.

There are several theoretical frameworks and flip learning models that have been developed based on the needs of various subjects [25]–[27], [29]. The Strayer [26] framework describes in general terms the potential of FC in the learning environment using technology. In comparison, the Strayer [27] framework has more details about knowledge building to students through activities and information gained by way of flip learning. The Murillo-Zamorano *et al.* [25] framework discusses various learning methodologies' flexibility and adaptability to make FC a special learning method, especially for new generation students. There are four main dimensions in the FC framework of higher education, namely knowledge, skills, satisfaction, and engagement, which can support the successful implementation of FC in learning. Based on the analysis of several studies that have been conducted, Lo and Hew [29] have proposed a basic model of FC. This model has a clearer description of extracurricular activities and classroom activities. Out-of-class learning activities occur at the level of remembering and understanding skills in the learning taxonomy, while in-class activities focus on higher levels of knowledge application skills and problem-solving at higher or difficult levels. This model also describes the support of online discussions outside the classroom and the support of teachers and peers in the classroom. For learning outside the classroom, this model suggests several activities that can be carried out in general. In general, the basic model of Lo and Hew [29] is suitable to be adopted as the framework of implementing FC in programming. The selection of that model is made based on the advantages and potential enhancements as listed in Table 5.

TABLE V
ADVANTAGES OF LO & HEW MODEL AND PROPOSED ENHANCEMENTS

Advantages	**Enhancements**
Suggestions for extracurricular and in-class activities are stated.	Out-of-class learning is broken down into pre-class and post-class sessions.
Skill levels for extracurricular and in-class activities are stated	The level of skills in learning outside and in the classroom is clearly stated.
Methods of support to students are suggested.	Appropriate activities are suggested in each learning session.
	Elements of motivation and control are included in learning activities.

The FC framework in programming is proposed based on the improvements of the basic model of FC by Lo and Hew [29]. The improvements made to this framework are also based on the findings from the analysis of the relevant articles that have been conducted. Through this framework, learning activities will occur in three phases: pre-class, in-class, and post-class. The learning objectives targeted at each phase are different. The level of knowledge in pre-class level learning activities is at the level of understanding and remembering. In the classroom phase, the learning objectives are targeted at building students' knowledge and skills. Next, in the post-classroom phase, the purpose of learning is to reinforce the new knowledge and skills possessed. There are also differences in the activities performed at each phase of learning. The main activities in the pre-class phase are watching videos, reading notes, and answering the quizzes provided. While in the classroom, active and collaborative learning activities are carried out, such as group discussions and presentations, and problem-solving. Program coding and testing activities are also carried out in the classroom according to the suitability of the learning topic. In the post-class phase, knowledge reinforcement activities are done through video reviews watched, individual coding exercises, completing assignments or quizzes.

Control is an aspect that needs to be considered in each phase involved, especially in the pre-class and post-class phases. In both phases, students will be doing the activities without the lecturer's supervision. In the pre-class phase, the control over students' involvement in activities is controlled through the log-in records into the learning system or portal. Quizzes are also used as a control method to ensure that students will watch the videos or read the necessary materials as prescribed in learning activities. The control aspect of the phase in the classroom is easier as there is a presence of the lecturer during the activity. Control can be made by monitoring students' involvement in the activities and attendance. Next, for the post-class phase, control aspects can be made by observing students' achievement in projects, work submitted online, or physical assignment results.

Motivation is the most important element in determining the success of a new approach introduced. Without motivation, students will face difficulties during the implementation of FC, especially when conducting the activities outside the classroom without supervision from the lecturer. Among the motivational elements used in the pre-class phase are awarding points to participation in activities or additional marks as an initiative to engage in activities. During the active learning activities in the classroom, immediate feedback from lecturers is used as motivation in the classroom. In the post-class phase, the motivation used is the same as the motivation in the pre-class phase. Fig. 5 is the proposed FC framework for the programming.

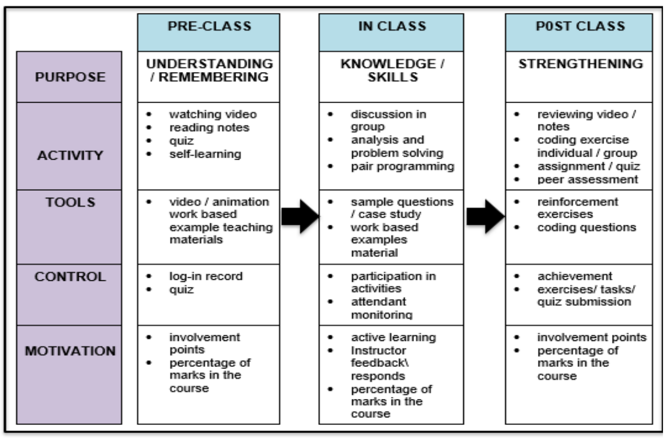

Fig. 5 The Proposed Framework of FC in Programming

IV. Conclusion

The implementation of FC in teaching and learning of programming can be the solution for some challenges and limitations that have been identified in programming subjects, such as time constraints, high cognitive load, and high dependence on lecturers. Some limitations in this study were identified as the analysis made was based on the selected research papers that abide by certain criteria. There are possibly other studies of flipped classrooms in programming that were not selected in the search. The limitation of existing frameworks and models of FC that are being used as benchmark for the proposed FC framework in programming. Further studies can be conducted to fill the content of the proposed framework to facilitate the implementation of FC as a strategy in programming.

Acknowledgment

The authors acknowledge Universiti Kebangsaan Malaysia and Faculty of Information Science and Technology for the research support. We also acknowledge the HLP scholarship provided by the Ministry of Education Malaysia.

References

[1] V. Renumol, S. Jayaprakash, and D. Janakiram, "Classification of cognitive difficulties of students to learn computer programming," *Indian Inst. Technol. India*, p. 12, 2009.

[2] N. Bubica and I. Boljat, "Strategies for Teaching Programming to Meet New Challenges: State of the Art," no. June 2014, 2015.

[3] B. Du Boulay, "Some Difficulties of Learning to Program," *J. Educ. Comput. Res.*, vol. 2, no. 1, pp. 57–73, 1986, doi: 10.2190/3LFX-9RRF-67T8-UVK9.

[4] H. Y. Durak, "Modeling Different Variables in Learning Basic Concepts of Programming in Flipped Classrooms," *J. Educ. Comput. Res.*, 2019, doi: 10.1177/0735633119827956.

[5] M. Ichinco and C. Kelleher, "Exploring novice programmer example use," *Proc. IEEE Symp. Vis. Lang. Human-Centric Comput. VL/HCC*, vol. 2015-Decem, pp. 63–71, 2015, doi: 10.1109/VLHCC.2015.7357199.

[6] A. Robins, J. Rountree, and N. Rountree, "Learning and Teaching Programming: A Review and Discussion Learning and Teaching Programming: A Review," vol. 3408, no. January, pp. 37–41, 2003.

[7] N. Shi, W. Cui, P. Zhang, and X. Sun, "Evaluating the Effectiveness Roles of Variables in the Novice Programmers Learning," *J. Educ. Comput. Res.*, vol. 56, no. 2, pp. 181–201, 2018, doi: 10.1177/0735633117707312.

[8] S. M. Shuhidan, M. Hamilton, and D. D'Souza, "Understanding novice programmer difficulties via guided learning," *Proc. 16th Annu. Jt. Conf. Innov. Technol. Comput. Sci. Educ. - ITiCSE '11*, p. 213, 2011, doi: 10.1145/1999747.1999808.

[9] S. Mohorovičić and V. Strčić, "An Overview of Computer Programming Teaching Methods," *Proc. 22nd Cent. Eur. Conf. Inf. Intell. Syst.*, pp. 47–52, 2011.

[10] A. Gomes and A. Mendes, "A teacher's view about introductory programming teaching and learning: Difficulties, strategies and motivations," *Proc. - Front. Educ. Conf. FIE*, vol. 2015-Febru, no. February, 2015, doi: 10.1109/FIE.2014.7044086.

[11] S. I. Malik and J. Coldwell-Neilson, "A model for teaching an introductory programming course using ADRI," *Educ. Inf. Technol.*, vol. 22, no. 3, pp. 1089–1120, 2017, doi: 10.1007/s10639-016-9474-0.

[12] E. Lahtinen, K. Ala-mutka, and H.-M. Jarvinen, "A Study of the Difficulties of Novice Programmers," *ACM ITiCSE'05*, pp. 14–18, 2005, doi: 10.1145/1151954.1067453.

[13] A. Vihavainen, J. Airaksinen, and C. Watson, "A systematic review of approaches for teaching introductory programming and their influence on success," *Proc. tenth Annu. Conf. Int. Comput. Educ. Res. - ICER '14*, pp. 19–26, 2014, doi: 10.1145/2632320.2632349.

[14] X.-M. Wang, G.-J. Hwang, Z.-Y. Liang, and H.-Y. Wang, "Enhancing Students' Computer Programming Performances, Critical Thinking Awareness and Attitudes towards Programming: An Online Peer-Assessment Attempt," *Educ. Technol. Soc.*, vol. 20, no. 4, pp. 1176–3647, 2017.

[15] S. Alhazbi, "Using flipped classroom approach to teach computer programming," *Proc. 2016 IEEE Int. Conf. Teaching, Assess. Learn. Eng. TALE 2016*, no. December, pp. 441–444, 2016, doi: 10.1109/TALE.2016.7851837.

[16] Y. Qian and J. Lehman, “Students’ Misconceptions and Other Difficulties in Introductory Programming,” *ACM Trans. Comput. Educ.*, vol. 18, no. 1, pp. 1–24, 2017, doi: 10.1145/3077618.
[17] G. Akçayır and M. Akçayır, “The flipped classroom: A review of its advantages and challenges,” *Comput. Educ.*, vol. 126, pp. 334–345, 2018, doi: 10.1016/j.compedu.2018.07.021.
[18] D. Pawelczak, “Comparison of traditional lecture and flipped classroom for teaching programming,” *Proc. 3rd Int. Conf. High. Educ. Adv.*, pp. 391–398, 2017, doi: 10.4995/HEAD17.2017.5226.
[19] M. Lundin, A. Bergviken Rensfeldt, T. Hillman, A. Lantz-Andersson, and L. Peterson, “Higher education dominance and siloed knowledge: a systematic review of flipped classroom research,” *Int. J. Educ. Technol. High. Educ.*, vol. 15, no. 1, 2018, doi: 10.1186/s41239-018-0101-6.
[20] M. N. Giannakos, J. Krogstie, and D. Sampson, “Putting Flipped Classroom into Practice: A Comprehensive Review of Empirical Research,” *Digit. Technol. Sustain. Innov. Improv. Teach. Learn.*, pp. 27–44, 2018, doi: 10.1007/978-3-319-73417-0_2.
[21] W. Kelly, “Flipping the Classroom to Solve the Time Problem,” 2017.
[22] F. H. Wang, “An exploration of online behaviour engagement and achievement in flipped classroom supported by learning management system,” *Comput. Educ.*, vol. 114, pp. 79–91, 2017, doi: 10.1016/j.compedu.2017.06.012.
[23] S. R. MD Derus and A. Z. Mohamad Ali, “Integration of Visualization Techniques and Active Learning Strategy In Learning Computer Programming: A Proposed Framework,” *Int. J. New Trends Educ. Their Implic.*, no. January, pp. 93–103, 2014.
[24] C. Stöhr and T. Adawi, “Flipped Classroom Research: From ‘Black Box’ to ‘White Box’ Evaluation,” *Educ. Sci.*, vol. 8, no. 1, p. 22, 2018, doi: 10.3390/educsci8010022.
[25] L. R. Murillo-Zamorano, J. Á. López Sánchez, and A. L. Godoy-Caballero, “How the flipped classroom affects knowledge, skills, and engagement in higher education: Effects on students’ satisfaction,” *Comput. Educ.*, vol. 141, no. October 2018, 2019, doi: 10.1016/j.compedu.2019.103608.
[26] J. F. Strayer, “The Effects of The Classroom Flip on The Learning Environment: A Comparison of Learning Activity In a Traditional Classroom And a Flip Classroom That Used An Intelligent Tutoring System,” Ohio State University, 2007.
[27] J. F. Strayer, “Designing Instruction for Flipped Classrooms,” in *322 - 349*, New York: Routledge, 2017, pp. 322–349.
[28] R. Talbert, *Flipped Learning - A Guide for Higher Education Faculty*. Sterling, Virginia: Stylus Publishing, LLC, 2017.
[29] C. K. Lo and K. F. Hew, “A critical review of flipped classroom challenges in K-12 education: possible solutions and recommendations for future research,” *Res. Pract. Technol. Enhanc. Learn.*, vol. 12, no. 1, p. 4, 2017, doi: 10.1186/s41039-016-0044-2.
[30] R. Brewer and S. Movahedazarhouligh, “Successful stories and conflicts: A literature review on the effectiveness of flipped learning in higher education,” *J. Comput. Assist. Learn.*, vol. 34, no. 4, pp. 409–416, 2018, doi: 10.1111/jcal.12250.
[31] A. Karabulut-ilgu, N. J. Cherrez, and C. T. Jahren, “A systematic review of research on the flipped learning method in engineering education,” *Br. J. Educ. Technol.*, vol. 00, no. 00, 2017, doi: 10.1111/bjet.12548.
[32] J. W. Baker, “The Origins Of ‘The Classroom Flip,’” *Proc. 1 St Annu. High. Educ. Flip. Learn. Conf.*, 2016.
[33] M. J. Lage, G. J. Platt, and M. Treglia, “Inverting the {Classroom}: {A} {Gateway} to {Creating} an {Inclusive} {Learning} {Environment},” *J. Econ. Educ.*, vol. 31, no. 1, pp. 30–43, 2000, doi: 10.1080/00220480009596759.
[34] M. Eric, *Peer Instruction A User’s Manual*, vol. 11, no. 3. 2000.
[35] B. Isong, “A Methodology for Teaching Computer Programming: first year students’ perspective,” *Int. J. Mod. Educ. Comput. Sci.*, vol. 6, no. 9, pp. 15–21, 2014, doi: 10.5815/ijmecs.2014.09.03.
[36] R. Yilmaz and H. Keser, “The Impact of Interactive Environment and Metacognitive Support on Academic Achievement and Transactional Distance in Online Learning,” *J. Educ. Comput. Res.*, vol. 55, no. 1, pp. 95–122, 2016, doi: 10.1177/0735633116656453.
[37] H. C. Lin and G. J. Hwang, “Research trends of flipped classroom studies for medical courses: a review of journal publications from 2008 to 2017 based on the technology-enhanced learning model,” *Interact. Learn. Environ.*, vol. 0, no. 0, pp. 1–17, 2018, doi: 10.1080/10494820.2018.1467462.
[38] W. J. Shyr and C. H. Chen, “Designing a technology-enhanced flipped learning system to facilitate students’ self-regulation and performance,” *J. Comput. Assist. Learn.*, vol. 34, no. 1, pp. 53–62, 2017, doi: 10.1111/jcal.12213.
[39] G. Hwang and P. Chen, “Effects of a collective problem-solving promotion- based flipped classroom on students’ learning performances and interactive patterns,” *Interact. Learn. Environ.*, vol. 0, no. 0, pp. 1–16, 2019, doi: 10.1080/10494820.2019.1568263.
[40] Z. Turan and B. Akdag-cimen, “Flipped classroom in English language teaching: a systematic review Flipped classroom in English language teaching : a systematic review,” *Comput. Assist. Lang. Learn.*, vol. 0, no. 0, pp. 1–17, 2019, doi: 10.1080/09588221.2019.1584117.
[41] T. Wang, “Overcoming barriers to ‘flip’: building teacher’s capacity for the adoption of flipped classroom in Hong Kong secondary schools,” *Res. Pract. Technol. Enhanc. Learn.*, vol. 12, no. 1, 2017, doi: 10.1186/s41039-017-0047-7.
[42] J. F. Strayer, “How learning in an inverted classroom influences cooperation, innovation and task orientation,” *Learn. Environ. Res.*, vol. 15, no. 2, pp. 171–193, 2012, doi: 10.1007/s10984-012-9108-4.
[43] A. Knutas, A. Herala, E. Vanhala, and J. Ikonen, “The Flipped Classroom Method: Lessons Learned from Flipping Two Programming Courses,” *Proc. 17th Int. Conf. Comput. Syst. Technol. 2016*, no. June, pp. 423–430, 2016, doi: 10.1145/2983468.2983524.
[44] M. Lou Maher, C. Latulipe, H. Lipford, and A. Rorrer, “Flipped Classroom Strategies for CS Education,” *Proc. 46th ACM Tech. Symp. Comput. Sci. Educ. - SIGCSE ’15*, pp. 218–223, 2015, doi: 10.1145/2676723.2677252.
[45] K. Krippendorff, *Content Analysis An Introduction to Its Methodology*, Second. London: SAGE Publications Inc, 2004.
[46] G. A. Bowen, *Document analysis as a qualitative research method*, vol. 9, no. 2. 2009.
[47] P. C. Sierra-Correa and J. R. Cantera Kintz, “Ecosystem-based adaptation for improving coastal planning for sea-level rise: A systematic review for mangrove coasts,” *Mar. Policy*, vol. 51, pp. 385–393, 2015, doi: 10.1016/j.marpol.2014.09.013.
[48] P. Cronin, F. Ryan, and M. Coughlan, “Undertaking a literature review: a step-by-step approach,” *Br. J. Nurs.*, vol. 17, no. 1, pp. 38–43, 2008, doi: 10.1177/107808747000500401.
[49] D. Morris and N. Ecclesfield, “A new computer-aided technique for qualitative document analysis,” *Int. J. Res. Method Educ.*, vol. 34, no. 3, pp. 241–254, 2011, doi: 10.1080/1743727X.2011.609547.
[50] J. R. Landis and G. G. Koch, “The Measurement of Observer Agreement for Categorical Data,” *Biometrics*, vol. 33, no. 1, p. 159, 1977, doi: 10.2307/2529310.
[51] O. S. Yan and G. Cheng, “Exploring the impact of flipped classroom on students’ acceptance of programming in secondary education,” *Proc. 2017 IEEE Int. Conf. Teaching, Assess. Learn. Eng. TALE 2017*, vol. 2018-Janua, no. December, pp. 246–249, 2017, doi: 10.1109/TALE.2017.8252341.
[52] S. Nikolic, M. Ros, and D. B. Hastie, “Teaching programming in common first year engineering: discipline insights applying a flipped learning problem-solving approach,” *Australas. J. Eng. Educ.*, vol. 00, no. 00, pp. 1–12, 2018, doi: 10.1080/22054952.2018.1507243.
[53] A. E. Chis, A.-N. Moldovan, L. Murphy, P. Pathak, and C. H. Muntean, “Investigating Flipped Classroom and Problem-based Learning in a Programming Module for Computing Conversion Course,” *J. Educ. Technol. Soc.*, vol. 21, pp. 232–247, 2018.
[54] J. Elmaleh, “Improving Student Learning in an Introductory Programming Course Using Flipped Classroom and Competency Framework,” in *IEEE Global Engineering Education Conference, EDUCON*, 2017, no. April, pp. 49–55.
[55] Y. Hayashi, K. Fukamachi, and H. Komatsugawa, “Collaborative Learning in Computer Programming Courses That Adopted The Flipped Classroom,” in *2015 International Conference on Learning and Teaching in Computing and Engineering Collaborative*, 2015, pp. 209–212, doi: 10.1109/LaTiCE.2015.43.
[56] T. B. Bati, H. Gelderblom, and J. van Biljon, “A blended learning approach for teaching computer programming: Design for large classes in Sub-Saharan Africa,” *Comput. Sci. Educ.*, vol. 24, no. 1, pp. 71–99, 2014, doi: 10.1080/08993408.2014.897850.
[57] B. J. Da Silva Estácio and R. Prikladnicki, “Distributed pair programming: A systematic literature review,” *Inf. Softw. Technol.*, vol. 63, pp. 1–10, 2015, doi: 10.1016/j.infsof.2015.02.011.
[58] J. Pattanaphanchai, “An Investigation of Students’ Learning Achievement and Perception using Flipped Classroom in an Introductory Programming course : A Case Study of Thailand Higher Education,” *J. Univ. Teach. Learn. Pract.*, vol. 16, no. 5, 2019.
[59] G. Akçayır and M. Akçayır, “The flipped classroom: A review of its advantages and challenges,” *Comput. Educ.*, vol. 126, no. January, pp.

334–345, 2018, doi: 10.1016/j.compedu.2018.07.021.
[60] K. Zeuch, S. Kaven, and V. Skwarek, "Evaluation of a re-designed introductory course" Programming in C " with video support," in *2019 18th International Conference on Information Technology Based Higher Education and Training (ITHET)*, 2019, pp. 1–6.
[61] M. J. D'Souza and P. Rodrigues, "Investigating the effectiveness of the flipped classroom in an introductory programming course," *New Educ. Rev.*, vol. 40, no. 2, pp. 129–139, 2015, doi: 10.15804/tner.2015.40.2.11.
[62] T. Ishak, R. Kurniawan, Z. Zainuddin, and C. M. Keumala, "The role of pre-class asynchronous online video lectures in flipped-class instruction: identifying students' perceived need satisfaction," *J. Pedagog. Res.*, pp. 1–11, 2019, doi: 10.33902/jpr. v4i1.145.
[63] H. Jonsson, "Using flipped classroom, peer discussion, and just-in-time teaching to increase learning in a programming course," *Proc. - Front. Educ. Conf. FIE*, vol. 2014, 2015, doi: 10.1109/FIE.2015.7344221.
[64] R. M. Ryan and E. L. Deci, "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being.," *Am. Psychol.*, vol. 55, no. 1, pp. 68–78, 2000, doi: 10.1037/0003-066X.55.1.68.
[65] J. Elmaleh and V. Shankararaman, "Improving student learning in an introductory programming course using flipped classroom and competency framework," *IEEE Glob. Eng. Educ. Conf. EDUCON*, no. April, pp. 49–55, 2017, doi: 10.1109/EDUCON.2017.7942823.
[66] L. Cao and M. Grabchak, "Interactive Preparatory Work in a Flipped Programming Course," in *ACM Global Computing Education Conference 2019*, 2019, pp. 229–235.
[67] C. Lih-shyang, C. Shu-Han, C. Chao-Cheng, and Y. Emily, "An e-learning system for programming language with semi-Automatic Grading," *10th Int. Conf. Ubi-media Comput. Work.*, 2017.
[68] I. Blau and T. Shamir-Inbal, *Re-designed flipped learning model in an academic course: The role of co-creation and co-regulation*, vol. 115. Elsevier Ltd, 2017.
[69] Y.-H. Chang, A.-C. Song, and R.-J. Fang, "Integrating ARCS Model of Motivation and PBL in Flipped Classroom: a Case Study on a Programming Language," *EURASIA J. Math. Sci. Technol. Educ.*, vol. 14, no. 12, 2018.
[70] M. Fetaji, B. Fetaji, C. Sukic, A. Gylcan, and M. Ebibi, "Case Study Analyses of the Impact of Flipped Learning in Teaching Programming Robots," *TEM J.*, vol. 5, no. 4, pp. 560–565, 2016, doi: 10.18421/TEM54-21.
[71] K. Thongkoo, P. Panjaburee, and K. Daungcharone, "Integrating inquiry learning and knowledge management into a flipped classroom to improve students ' web programming performance in higher education," *Knowl. Manag. E-Learning*, vol. 11, no. 3, pp. 304–324, 2019.
[72] L. Chen, S. Chang, C. Chen, and E. Yang, "An e-Learning system for programming languages with semi-automatic grading," *10th Int. Conf. Ubi-media Comput. Work. An*, 2017.
[73] M. Hai and H. Li, "Research and Application of Project-based Teaching Reform Method Based on Flipped Classroom in the Teaching of Programming Design Courses," in *IInternational Conference on Mechanical and Energy Technologies 2019*, 2019, pp. 49–53.
[74] A. Mohamed, "Designing a CS1 Programming Course for a Mixed-Ability Class," in *WCCCE '19*, 2019, pp. 10–15.
[75] Y. Chang, A. Song, and R. Fang, "The Study of Programming Language Learning by Applying Flipped Classroom," *2018 1st IEEE Int. Conf. Knowl. Innov. Invent.*, no. 1, pp. 286–289, 2018.
[76] O. S. Yan and G. Cheng, "Exploring the impact of flipped classroom on students' acceptance of programming in secondary education," *Proc. 2017 IEEE Int. Conf. Teaching, Assess. Learn. Eng. TALE 2017*, vol. 2018-Janua, no. December, pp. 246–249, 2018, doi: 10.1109/TALE.2017.8252341.
[77] V. Gupta, "Blended SPOC Teaching and Learning Model for Computer Programming Course: Insights and Defeating Challenges," *IEEE TALE2020 - An Int. Conf. Eng. Technol. Educ.*, pp. 251–257, 2020.
[78] H. Y. Durak, "Flipped learning readiness in teaching programming in middle schools: Modelling its relation to various variables," *J. Comput. Assist. Learn.*, no. July, pp. 939–959, 2018, doi: 10.1111/jcal.12302.
[79] Z. Zainuddin, H. Haruna, X. Li, Y. Zhang, and S. K. W. Chu, "A systematic review of flipped classroom empirical evidence from different fields: what are the gaps and future trends?," *Emerald Insight- Horiz.*, p. OTH-09-2018-0027, 2019, doi: 10.1108/OTH-09-2018-0027.
[80] S. Djenic, R. Krneta, and J. Mitic, "Blended learning of programming in the internet age," *IEEE Trans. Educ.*, vol. 54, no. 2, pp. 247–254, 2011, doi: 10.1109/TE.2010.2050066.
[81] S. Mohorovic and E. Tijan, "Blended learning model of teaching programming in higher education," *Int. J. Knowl. Learn.*, vol. 7, no. 1/2, p. 86, 2011, doi: 10.1504/IJKL.2011.043893.
[82] S. Alhazbi, "Active Blended Learning to Improve Students' Motivation in Computer Programming Courses : A Case Study," in *Advances in Engineering Education in the Middle East and North Africa*, 2016, pp. 187–204.
[83] S. An, W. Li, J. Hu, L. Ma, and J. Xu, "Research on the Reform of Flipped Classroom in Computer Science of University Based on SPOC," in *The 12th International Conference on Computer Science & Education (ICCSE 2017)*, 2017, no. ICCSE, pp. 621–625.
[84] A. Amresh, A. R. Carberry, and J. Femiani, "Evaluating the effectiveness of flipped classrooms for teaching CS1," *Proc. - Front. Educ. Conf. FIE*, pp. 733–735, 2013, doi: 10.1109/FIE.2013.6684923.
[85] B. A. Ali Aljaani and M. A. Yousuf, "Flipping Introductory Engineering Design Courses : Evaluating Their Effectiveness," in *IEEE Global Engineering Education Conference*, 2016, no. April, pp. 234–239.
[86] H. Özyurt and Ö. Özyurt, "Analyzing the effects of adapted flipped classroom approach on computer programming success, attitude toward programming, and programming self-efficacy," *Comput. Appl. Eng. Educ.*, vol. 26, no. 6, pp. 2036–2046, 2018, doi: 10.1002/cae.21973.