Flipping the Foundation: A Multi-Year Flipped Classroom Study for a Large-Scale Introductory Programming Course

Ms. Emily Ann Marasco, University of Calgary

Emily Marasco is a Ph.D. candidate in Electrical Engineering at the University of Calgary. Her research focuses on interdisciplinary creativity and how creative thinking development can be integrated within technical STEM learning. Emily is a co-founder of Learning Academy for Teaching Techniques and Interdisciplinary Curriculum Enhancement (LATTICE) Development, and is involved in educational technology development initiatives. She is the Canadian Engineering Memorial Foundation 2016 Claudette MacKay-Lassonde Award recipient for her community and outreach accomplishments in inspiring women in engineering.

Dr. Mohammad Moshirpour, University of Calgary

Dr. Mohammad Moshirpour is an instructor of Software Engineering at the Schulich School of Engineering, University of Calgary. His research interests are the areas of software architecture, software requirements engineering, design and analysis of multi-agent systems, intelligent data analytics, data mining and machine learning, and software engineering and computer science education. He is the IEEE Chair of the Computer Chapter of the Southern Alberta Section.

Dr. Mahmood Moussavi, University of Calgary

Dr. Mahmood Moussavi is currently a faculty member of the Department of Electrical and Computer Engineering, at the University of Calgary. He has over 25 years research and teaching experience in different areas of software engineering and information technology, and he has served over 11 year as Director of the Software Engineering at the above-mentioned department.

His general research interest and expertise are summarized as follows: Distributed and multi-agent software systems Software design and architecture Data mining and database management

Dr. Moussavi's most recent research projects involves: A simulation-based study on software design for connected vehicles and traffic infrastructures, applying agent-based approach. Development of a stochastic time series generator with adaptive software architecture A study on detecting emergent behaviours in distributed systems using an ontology based methodology.

Flipping the Foundation: A Multi-Year Flipped Classroom Study for a Large-Scale Introductory Programming Course

Abstract

The flipped classroom format is becoming increasingly accepted among many engineering schools. By offering traditional lecture material online or outside of the scheduled class time, students are able to spend their interaction time with the instructor on hands-on homework and application activities.

Like many undergraduate engineering programs, at the Schulich School of Engineering all first year students are required to take the introductory computer programming course. With course enrollment around 800 students, and the relatively wide variety of students with different programming backgrounds, it would be challenging to provide the students with an in-depth learning experience by relying only on a traditional lecture hall format. A flipped classroom provides students with the opportunity to explore the implementation of theoretical programming concepts previously learned online.

After a significant redesign, this course has been run in a flipped classroom format since summer of 2015. Weekly video lecture content is provided through an online learning management system, and students are held accountable through embedded quizzes. Students collaborate on interactive studio exercises both inside and outside of class time, culminating in a final creative design project.

Data regarding student performance, assessment, and perceptions has been collected over several semesters and will be analyzed and presented in the final paper. Both quantitative and qualitative measures will be included in a mixed methods study. Student performance in consecutive courses will also be discussed in comparison to previous non-flipped cohorts. This paper will outline the design and implementation over two years for this large-scale flipped introductory programming course, and will provide recommendations to other educators based on experience and lessons learned.

Introduction

The flipped classroom format has become increasingly popular among engineering courses [Velegol, 2015]. Technical instruction and content are provided online or via another method external to scheduled course time. This allows the instructor to use their face-to-face time with the students to facilitate exploratory learning and hands-on application for more in-depth understanding [Kolb, 2005] [Robinson, 2004]. Various forms of flipped classrooms have been tried in electrical, computer, and software engineering education, from early low-tech activities [Bland, 2006] to videos with programming exercises [Gannod, 2007] [Talbert, 2012]. However, there are few reports on flipped classrooms for large cohorts [Velegol, 2015], which present their own challenges for scalability and feasibility. This paper will provide an overview of a flipped classroom implementation for a large-scale first year programming course, feedback from several semesters of implementation, and revisions based on lessons learned, which may be helpful for educators in similar circumstances.

The first year introductory programming course at the Schulich School of Engineering is mandatory for all first year engineering students, regardless of their intended specialization. This common core course, identified as ENGG 233: Computer for Engineers is part of the foundation program that allows students to explore various engineering fields while learning the required core technical concepts. The course is typically run during a 13-week fall semester, and in recent years have had an enrollment total averaging 800 students. A second offering is made available in the spring or summer semesters, usually with a much smaller class size.

Prior to July 2015, ENGG 233 followed a traditional lecture format. Content was delivered during three one-hour lectures each week in a large theater-style format. Students practiced their application skills in C++ programming during a two-hour weekly laboratory period with guidance from graduate student teaching assistants. In 2015, the faculty decided to redesign the course with emphasis on algorithmic thinking and exploratory, applied learning [Pears, 2007]. The language of focus was changed to Processing, a Java-based language that allows novice coders to produce visual graphics output. This visual component allows students to apply algorithmic logic solutions for engaging, but not complex, creative graphical exercises.

With the engineering students already facing timetable restrictions and a demanding course load, the programming course was changed to a flipped format to help facilitate more hands-on time with an instructor. Similar to learning a second spoken language, computer programming education requires application and problem-solving situation to better understand vocabulary and syntax structure. Students enrolled in ENGG 233 are considered to be completely new to programming, and no pre or co-requisite courses are necessary. The course covers the following concepts:

- Introduction to computer structure, machine code, and algorithms
- Variables, arithmetic, operations, char & string data types
- Decision structures and basic control flow
- Repetition structures
- Functions
- Arrays

- Introduction to classes and objects

To further implement facilitation of algorithmic thinking concepts [Pears, 2007] and creative application, the following concepts were added to the flipped offering as part of the hands-on exercises and final design project.

- Understanding of graphical interface and user interaction.
- Designing for a target audience and program testing.
- Algorithmic and modular thinking, from a detailed understanding of functions (smallest program unit) to a brief introduction to objects and classes.
- Sustainability issues in ICT, including software and hardware issues impacting nature, the economy, society, and individual well-being.

The following sections will focus on the implementation of the flipped classroom format across multiple offerings of ENGG 233.

Initial Implementation

Instructional content was posted weekly using an online learning management system. Each content release included several videos developed by the instructors, averaging ten to fifteen minutes in length with embedded quiz questions. While these quizzes were worth very little of the overall grade, the weekly deadline provided students with a form of accountability. The video content varied depending on the topic, and included a combination of PowerPoint slides, real-time handwritten notes and real-time screen-captured coding. Additional tutorials were occasionally posted to provide in-depth exploration of particularly difficult topics, and as well as links to external media and material. Each content release package was posted one week prior to its corresponding laboratory session. A pilot course as offered in the summer of 2015 to test any technical or logistical issues before the course was offered on a large-scale basis. Student performance in the pilot course was comparable to the traditional format, and no major challenges were presented.

Prior to the first large-scale offering in Fall 2015, the hands-on instructor-led portion of the course was modified for scalability. To better facilitate the large studio laboratory sessions (200 students in each scheduled session), graduate teaching assistants were trained to manage the hands-on, exploratory exercises in a studio format. Each 200-student session was divided into six sessions running concurrently, each with a designated graduate teaching assistant. One graduate coordinator and two instructors were also available, and instructors were able to communicate with the entire student base via an overhead audio system. Adapted from fine arts classes, students were seated in groups while working on individual assignments to promote collaboration and peer assistance. Instructors and graduate students provided guidance during in-studio exercises, and students were also assigned post-studio work, due during the following week.

The course concluded with a large design project that combined technical course content with creative thinking. Students were asked to design and program their own interactive game while fulfilling a set of technical criteria. Students were also free to work individually or with a partner, and no restriction was set on the theme or type of game. This course format was offered again in spring 2016 to a much smaller class size. The spring class is offered primarily for students who did not successfully complete the fall session, or for some legitimate reason missed the previous session(s).

Following the conclusion of semester, students were asked to provide feedback via a quantitative and qualitative survey. The results of this survey were used to inform changes for the following year. The 2015 and 2016 survey results are presented together later in this paper.

Implementation Revisions

Following the experiences of the first year, student feedback and perceptions were reviewed regarding the flipped classroom format, blended learning techniques, and project-based approaches. Several changes were made for the Fall 2016 offering of ENGG 233.

Revision #1: Weekly In-Class Tutorial

During the 2015 semester, no in-class lectures or tutorial sessions were offered and all of the course content was delivered via lecture videos. Students relied on the weekly studio sessions for help from their instructor or graduate teaching assistant, or they had to seek help from the general engineering tutoring centre.

For 2016, the delivery of the course was changed to add a tutorial component. Subsequently, in addition to attending the weekly laboratory sessions, students also attended a 75-minute tutorial each week. As in 2015, course content was posted online in video format, and no new content was introduced during in-class tutorials. Instead, tutorials served as instructor-led problem-solving sessions, allowing instructors to better gauge real-time student comfort with example material.

Each tutorial began with the instructor providing a brief overview of the posted videos and other supplemental content, followed by the instructor solving a number of practice problems in detail (ranging from simple to challenging, including past midterm and final exam questions). The tutorials also provided the opportunity for the students to ask questions, and allowed the instructor to construct a more comprehensive view of the students' progress.

Revision #2: Modifying the Project

By the end of the term, students learn a variety of programming constructs which allow them to create practical software programs. During the 2015 version of the course, the students were given a list of categories such as game design, sustainability-related solutions, and data visualization, and were asked to propose, design and develop their own software program. In 2016, student enrollment rose to approximately 900, so the scope of the project was decreased to one particular theme- a modified version of a pool-table game. Implementing this change reduced the wide variety of solutions, therefore increasing the manageability and amount of technical support and trouble-shooting provided for the students. This change also helped to standardize the level of difficulty between the projects. To emphasize the creativity aspects of the project, students were still given the choice to design and develop their own particular features in the game.

A qualitative and quantitative feedback survey was again run following the conclusion of the course.

Data-Based Recommendations

Overall, two years of running the flipped classroom format for introductory programming has been successful. However, this initiative is continually under revision as needed to facilitate improved student learning.

The performance of the students in the flipped classroom format was comparable to previous traditional format offerings. In 2015, the final student average was a B+, and the 2016 final student average was between B/B+. This compares well to the previous 2014 average of a B grade. However, it should be noted that the grade breakdown varied between each offering:

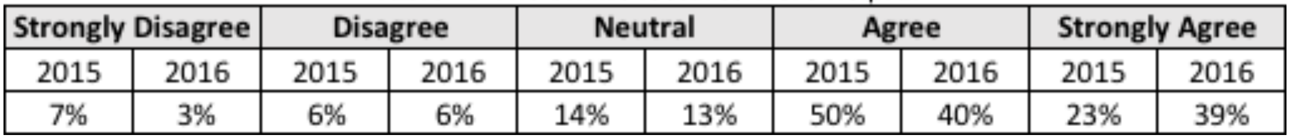
- In 2014: 50% final exam, 30% midterm exam, and 20% lab assignments.
- In 2015: 40% final exam, no midterm exam, 15% quizzes, and 45% lab/project assignments.
- In 2016: 40% final exam, 25% midterm exam, 5% quizzes, and 30% lab/project assignments.

The grade breakdown was changed based on 2015 implementation results. The lack of midterm exam was originally intended to provide students with more time and emphasis on the creative design project. However, students reported increased anxiety regarding the final exam format without first taking a midterm, and instructors preferred having a comprehensive mid-semester evaluation of the students' technical progress. The emphasis on quizzes was reduced in 2016 to provide adequate valuation within the midterm and lab/project categories. The 2015 results revealed that students were motivated to answer the weekly quizzes as long as some marks were given, regardless of the minimal overall grade contribution.

Students were asked to complete a post-course survey that combined qualitative open-ended responses and quantitative rankings on a five-point Likert scale. This data has received on-going analysis to compare the 2015 and 2016 cohorts, and to provide insight into any future changes.

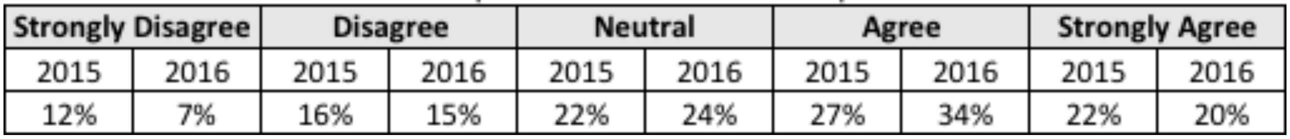
Perceptions of online lectures: In 2015, 73% of students agreed or strongly agreed that the online lectures were helpful, and 79% agreed in 2016. When asked if they prefer online lectures to in-person lectures, such as those in their other engineering courses, approximately half of both the 2015 and 2016 cohorts agreed they would prefer online lectures. This was also reflected in the qualitative responses, where an average of half the respondents expressed a preference for the online lecture format over the in-person format. These results are summarized in Tables 1 and 2.

Table 1: The online lectures were helpful



Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
7%	3%	6%	6%	14%	13%	50%	40%	23%	39%

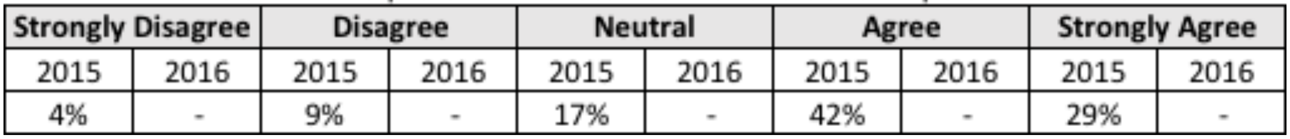
Table 2: I would prefer online lectures to in-person lectures



Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
12%	7%	16%	15%	22%	24%	27%	34%	22%	20%

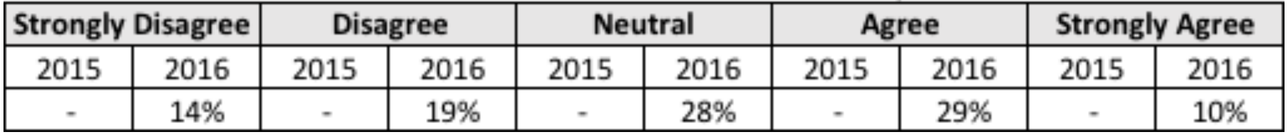
Addition of tutorials: As mentioned earlier, an instructor-led tutorial was introduced in 2016, due to instructor observations the previous year regarding the need for increased real-time feedback and interaction. This was also reflected in student feedback when students were asked if they would prefer to have in-personal tutorials in addition to the online lectures, with 71% agreeing. In 2016, 49% of students agreed that they preferred the in-person tutorials to the additional online tutorials that were posted, but only 39% agreed that the classroom tutorials were helpful. More information is needed to better understand these results. In-person tutorials may be more beneficial to weaker students, while stronger students may find the in-person reinforcement of video content to be boring or unnecessary. As well, some students may learn better at a slower pace with the opportunity to watch the video content several times, while also preferring the additional reinforcement of instructor-led repetition and examples in real-time. Initial results related to the tutorials (prospective in 2015, actually run in 2016) are found in Tables 3, 4, and 5, and additional investigations may be performed in the future.

Table 3: I would prefer online lectures with additional in-person tutorials



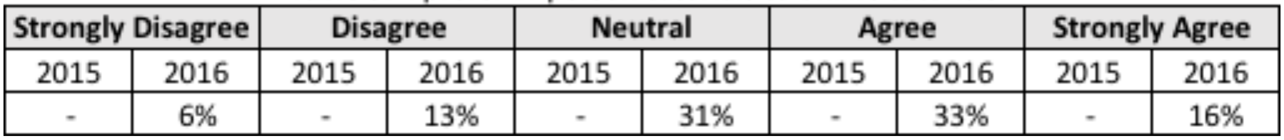
Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
4%	-	9%	-	17%	-	42%	-	29%	-

Table 4: The classroom tutorials were helpful



Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
-	14%	-	19%	-	28%	-	29%	-	10%

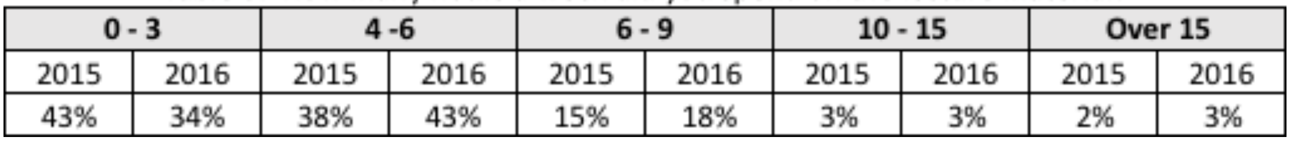
Table 5: I prefer in-person tutorials to online tutorials



Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
-	6%	-	13%	-	31%	-	33%	-	16%

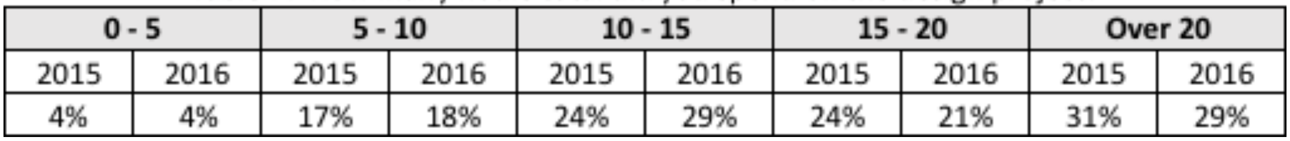
In general, the course workload did not differ between either cohort. The majority of students self-reported spending an average of three hours a week on lecture material. Since intent of the online lecture material was to replace the traditional three one-hour lectures each week, this was considered to be a positive report, and is shown in Table 6. Student effort and time spend on homework and problem-solving exercises varied week to week, but the majority of the students reported spending an average of 15 hours on the design project over a roughly three-week period, shown in Table 7.

Table 6: How many hours a week did you spend on the lecture material?



0 - 3		4 -6		6 - 9		10 - 15		Over 15	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
43%	34%	38%	43%	15%	18%	3%	3%	2%	3%

Table 7: How many hours total did you spend on the design project?



0 - 5		5 - 10		10 - 15		15 - 20		Over 20	
2015	2016	2015	2016	2015	2016	2015	2016	2015	2016
4%	4%	17%	18%	24%	29%	24%	21%	31%	29%

Students were also asked to provide feedback regarding the studio/laboratory portion of the flipped class. The majority of the students agreed or strongly agreed that their understanding of technical concepts was increased through the creative design project, as well as their creative thinking. A reflection on the 2015 studio cohort was published in the Proceedings of the 12th International CDIO Conference [Marasco, 2016]. Further analysis comparing the 2015 and 2016 projects is currently underway, and preliminary results show that students showed a slight preference for being able to choose their own project topic. However, this flexibility presents difficulties for grading consistency and technical support, and will be a topic of further research.

Students responded well overall to the concept of a design project, and reported value in both programming their own project and playtesting other groups' projects. One student provided feedback as:

> *"I believe that the game project allowed students to be creative and innovative. This project was very open ended which I believe was an advantage. Ultimately, I believe that this course was very well executed and organized. This course requires individuals to become better problem solvers, not just plug and chug."*
>
> *- ENGG 233 F15 Student*

In conclusion, the ENGG 233 flipped classroom initiative has been successfully implemented over two years and four different offerings: two large-scale offerings and two smaller spring/summer sessions. Students have performed consistently and according to past expectations, and a preliminary study of the 2015 students has shown that students were adequately prepared for the consecutive courses that followed in their chosen engineering program of study. Summarized recommendations from this paper include the implementation of an instructor-led tutorial to demonstrate additional examples that reinforce the online lecture learning, and the need for trained support to help facilitate creative design exercises among large classes. Additional analysis is planned for this initiative as the ENGG 233 continues to improve engineering education for first year introductory programming.

References

L. Bland. (2006). Applying flip/inverted classroom model in electrical engineering to establish life-long learning. American Society for Engineering Education Annual Conference. Chicago, Illinois.

G. C. Gannod. (2007). Work in Progress – Using podcasting in an inverted classroom. IEEE Frontiers in Education Conference. Milwaukee, Wisconsin.

A. Kolb and D. Kolb. (2005). Learning styles and learning space: Enhancing experiential learning in higher education. Academy of Management Learning & Education, 4(2):193-212.

E. Marasco, M. Moshirpour, L. Behjat, and W. Rosehart. (2016). Studios & sustainability: a creative CDIO approach to computer engineering education. 12th Annual International CDIO Conference. Turku, Finland.

A. Pears, S. Seidman, L. Malmi, L. Mannila, and E. Adams. (2007). A survey of literature on the teaching of introductory programming. Working Group Report on ITiCSE on Innovation and Technology in Computer Science Education. ACM, New York, NY, USA.
DOI: https://doi.org/10.1145/1345443.1345441

K. Robinson, and A. M. Azzam. (2009). Why creativity now? (interview). Educational Leadership, 67(1):22-26.

R. Talbert. (2012). Learning MATLAB in the inverted classroom. American Society for Engineering Education Annual Conference. San Antonio, Texas.

S. B. Velegol, S. E. Zappe, and E. Mahoney. (2015). The evolution of a flipped classroom: evidence-based recommendations. Advances in Engineering Education, Winter 2015.