Flipping Traditional CS Education Upside Down: A Study of Interventions in Two Core Computing Courses

Celine Latulipe, Bruce Long, Mary Lou Maher, Audrey Rorrer, Karen Bean
College of Computing and Informatics
Center for Education Innovation

Abstract

At UNC Charlotte, the Department of Software and Information Systems is embarking on a mission to revolutionize teaching in an ongoing effort to improve student learning outcomes and retention. We have incorporated promising teaching methodologies into a revised classroom structure for two core courses in the Department: introductory programming and introductory web programming. The overarching objective of restructuring these courses into a "flipped" classroom experience using complimentary teaching methods is to increase student engagement and performance. We believe that by accomplishing this central goal, a secondary goal of increasing student retention from freshmen to sophomore year can be achieved. While we have begun these pedagogical reforms, we have yet to conduct assessment of student performance and evaluation of these initiatives. The purpose of this proposal is to conduct a robust evaluation of these interventions, so that we may glean understanding about what methodologies are central to improving student learning and student retention. The intellectual merit of this project is in testing new teaching methods that have been demonstrated as effective elsewhere, developing a deeper understanding of how these methods impact our students, and in identifying methods that are viable approaches to infuse across our curriculum. These findings will enable us to enhance educational practices so that students benefit directly. The benefits to the College are improved student learning outcomes, retention and better academic performance overall. University benefits include quality enhancement including better student preparation for higher-level courses. The larger community benefits from knowledge of contextual factors pertaining to pedagogical innovations.

Budget Request for SOTL Grant
Year 2014

Joint Proposal? ____ Yes X No

Title of Project: **Flipping Traditional Education Upside Down: A Study of Interventions in Two Core Computing Courses**

Duration of Project: 18 months

Primary Investigator(s): Celine Latulipe, Bruce Long, Mary Lou Maher, Audrey Rorrer, Karen Bean

Email Address(es): clatulip@uncc.edu; nblong@uncc.edu; m.maher@uncc.edu; audrey.rorrer@uncc.edu; karen.bean@uncc.edu

UNC Charlotte SOTL Grants Previously Received (please names of project, PIs, and dates): None

Allocate operating budget to Department of: Software and Information Science

		Year One	Year Two
Account #	**Award**	**January 2014 to June 2014**	**July 2014 to December 2014**
Faculty Stipend	Transferred directly from Academic Affairs to Grantee on May 15	$ -	
911250	Graduate Student Salaries	10,000	10,000
911300	Special Pay (Faculty on UNCC payroll other than Grantee)		
915000	Student Temporary Wages		
915900	Non-student Temporary Wages		
920000	Honorarium (Individual(s) not with UNCC)		
921150	Participant Stipends		
925000	Travel - Domestic		5,000
926000	Travel - Foreign		
928000	Communication and/or Printing		
930000	Supplies		
942000	Computing Equipment		
944000	Educational Equipment		
951000	Other Current Services		
	GRAND TOTAL		**$25,000**

Attachments:

1. Attach/provide a narrative that explains how the funds requested will be used.

Implementation of the course interventions will be conducted by course instructors, Celine Latulipe and Bruce Long, as a part of their faculty role. Assessment and evaluation analyses will primarily be undertaken by a graduate student in the SIS Department, and overseen by Mary Lou Maher, CEI Director, and Audrey Rorrer, CEI Evaluator. The graduate student stipend total of $20,000 covers the normal rate paid for an academic year assistantship in the College. The graduate student salary will support 20 hours a week during the spring, summer and fall terms of 2014, when the student will perform data consolidation and preparations, data analyses, and assisting with writing methods and results components for dissemination (including presentations and papers). Qualitative study components will be conducted by Audrey Rorrer and Karen Bean, CEI Manager. Dissemination of study outcomes will include travel for 2 faculty members to the Association of Computer Science Special Interest Group: Computer Science Education. Estimated travel costs are based upon 2014 conference registration [$400 each= $800] and host city hotel fees at the conference group rate [$160 for 3 nights, per person= $960]. The 2015 conference will be held in Los Angeles, for which a current round trip flight costs [$1,200 per person]. This is a travel subtotal of $4,160 for which we have rounded up to $5,000 to account for expected increase in hotel and airfare by 2015.

2. Has funding for the project been requested from other sources? ___ Yes __X__ No. If yes, list sources.

UNC CHARLOTTE
College of Computing and Informatics
Office of the Dean
9201 University City Boulevard, Charlotte, NC 28223-0001
t/ 704.687.8450 f/ 704.687.6979 www.cci.uncc.edu

MEMORANDUM

TO: Scholarship of Teaching and Learning Grants Committee

FROM: Yi Deng, Dean

DATE: November 5, 2013

RE: Support for Proposal: *Flipping Traditional Education Upside Down: A Study of Interventions in Two Core Computing Courses*

I am pleased to offer my support for the SOTL proposal submitted by the interdisciplinary Center for Education Innovation in our College. This proposal enables support of rigorous study of newly deployed educational methods in introductory programming courses, in which far too many students are unsuccessful. With a drop, withdrawal and failure average rate of roughly 39% over the last few years in our introductory courses, this project has the potential to improve student learning outcomes in Introductory Programming (ITIS 1212) and in Web Applications (ITIS 2300), which is anticipated to promote and support persistence in studying computing. The research findings from this proposal will inform our college wide educational approach, as well as contribute to the academic knowledge at the national level. The College of Computing and Informatics is part of a national initiative to broaden the computing pipeline through multiple efforts that include pedagogy and research initiatives to support increased persistence and success in STEM areas like computer science and software information science. The proposed project is closely aligned with the goals of our Center for Education Innovation, and I am pleased to support this effort.

Sincerely,

Yi Deng, Ph.D.
Dean and Professor

Project Narrative

A. Specific Aims

In the College of Computing and Informatics, we face several challenges that are specifically associated with students in our introductory programming courses. We have adopted promising teaching methods to develop programming skills in novel ways, and propose a study to understand the effect of these interventions on students with respect to contextual factors. Our overall aims for this project are to study innovative CS education methodologies, deploy those most promising methods throughout the coursework in the College, and disseminate our findings and practices to the larger national community. The specific project objectives are to study the impact of new teaching methods in two introductory programming courses in the Software and Information Sciences Department (SIS) on immediate and longitudinal student learning outcomes and in retention in the major. Expected outcomes include increased student learning as indicated by course performance and increased retention in the major.

This project seeks to confirm the efficacy of new teaching methods within gateway programming courses, which have been stumbling blocks for far too many students historically. The introductory courses offered in the SIS Department, ITIS 1212 Introduction to Media Programming, and ITIS 2300 Web-Based Application Development, enroll approximately 200 students combined each semester, most of whom are first year students, i.e. freshmen and transfer students. These students arrive in these courses with various levels of pre-course exposure to programming, and college readiness in general. It is not surprising that the percentage of students receiving D, F, and W grades is has been historically high within our introductory programming courses, an average of 39% for ITCS 1212 (Programming I). Many students have to repeat gateway courses, and those with low performance are ill prepared for higher level course work. The impact on students is an increase in the number of credit hours taken, increased time to graduate, increased tuition, and a loss of full time income. The impact on faculty is course sections that are filled to capacity, which costs departmental resources.

To address these challenges, the SIS Department did two things. First, the department introduced ITIS 1212, an alternate version to the ITCS 1212 introductory programming course. ITIS 1212 uses a media computation approach to teaching programming. Second, the SIS Department began implementing new teaching methods to increase student learning outcomes in the two gateway courses, ITIS 1212 and ITIS 2300, to improve student performance over time. We have been unable to study the impact of these interventions and to make subsequent formative recommendations to the College, therefore, we propose a mixed-method design study to investigate how these interventions work, to what degree, and under what contextual factors across a heterogeneous study body. Students will gain enhanced educational engagement, furthering their ability to succeed in their academic pursuits. Faculty will gain knowledge of pedagogical practices that can enhance their teaching practice. Introductory programming courses serve over 500 students across the College each semester. Long term outcomes are improving quality of CS education, reduction of repeated courses, improved performance in higher-level follow-up courses, shortened time to graduate, and lowered course deliver costs.

There are two primary research questions that guide this project. First, do the interventions in introductory programming courses improve student learning in those courses? Sub-questions include how do these interventions impact students based upon identified demographic traits, such as gender, prior computing exposure, and incoming academic factors? Second, do these interventions improve student performance in subsequent courses, like programming II (ITCS 1213), and if so, for which students? The goals are to confirm the efficacy of these teaching methods both within the introductory courses and over time in preparing students for the next level of learning. Additionally, the study will enable discovery of what variability in outcomes may exist. This project has the potential to confirm prior findings regarding best practices in computing education, and to explore how these practices impact students by contextual factors. What is novel about our teaching practice is the study of these new methods with specific populations (women, prior preparation levels) and the longitudinal impact (year to year retention, performance in subsequent core courses).

B. Literature Review

Despite the national need for computer science graduates and call out by the President for one million more STEM graduates over the next decade (Office of Science and Technology Policy, 2012), CS education continues to see high failure rates in gateway courses and methodology entrenched in traditional pedagogy. Failure rates in introductory programming courses are generally between 30-50% worldwide (Bennedsen and Caspersen, 2007). Some have argued that traditional introductory computer programming education fails because it is antisocial, boring, and irrelevant instead of presented as socially relevant and cooperative (Porter, Guzdial, McDowell, and Simon, 2013). As a result, the number of alternative teaching methods being explored in CS education is increasing, with many promising approaches. Active learning, online learning, hybrid learning, and social learning are just a few of the popular methods being discussed, tried, and adapted for teaching CS. Yet the effect of these new teaching methods, along with a deeper understanding of the impact of interventions on specific student populations, remains unclear.

Active learning is unarguably more effective than passive learning done via lecture format. For example, Roediger and Karpicke's (2006) findings indicate that students are engaged in the most effective learning when they are actively retrieving from memory versus listening to a lecture. New approaches to teaching programming, such as the flipped classroom method, provides active learning and student engagement by using class time to solve problems, write code, and collaborate with their peers. Educators have used the flipped classroom technique successfully in a variety of subjects, including Computer Science (Kane and Fiedler, 2005), whereby students view lecture type materials outside of class so that the classroom can become an arena for active engagement and direct participation in structure activities. Studies have shown that working with peers in class programming activities improves overall learning, increases confidence and makes coding fun (Gehringer and Peddycord, 2013; Hanks, 2006; Lockwood and Esselstein, 2013). Pair programming is a demonstrated technique for successfully enhancing learning, particularly for women, and particularly when pairs are matched by gender (Bevan, Werner, & McDowell, 2002; Werner, Hanks, & McDowell, 2004). New technologies including 'Clickers' have successfully been used to foster learning by traditional Socratic method with a modern twist (Mayer, Stull, Ameroth, et al., 2009).

This shift in teaching CS to more active learning, particularly for courses that include programming, has the potential to address many of the learning and retention challenges faced in the traditional educational approach of a lecture format followed by individual lab work. We believe that the interest in adopting new teaching methods in CS education will be spurred by supplying pedagogical research demonstrating the conditions under which successful short term and long term student outcomes are most likely.

C. Methods

The overall project plan is to collect archival data on the course grade outcomes prior to the implementation of new teaching methods and conduct a comparative study of outcomes following the interventions. We began teaching interventions in Spring 2013 for ITIS 2300 and designed a new introductory computing course in Fall 2013, ITIS 1212. The following is a list of interventions that we have deployed in the ITIS 1212 and 2300 courses:

- **Flipped classroom**: Before coming to programming labs, students watch videos that teach core programming concepts, and complete assigned reading and practice programming.
- **Pair Programming**: Students engage in pair programming in the lab, and in gender matched pairs, with rotating partnerships to allow peer connectivity.
- **Media Computation**: Students learn programming through manipulation of highly engaging and visual multi-media elements and are encouraged to use their own images to demonstrate proficiency in personally unique ways.
- **Question Tokens**: Students are given special 3D printed tokens to exchange for posing a question to the instructor, thus prompting students to be self-sufficient learners.
- **Peer Instruction Workshops**: Peer instruction workshops are held in lieu of lectures.
- **Lightweight Teams**: Students are divided into teams of 5 at the beginning of the semester and sit with these teams during the peer instruction workshops each week, in assigned seating. They work together on quizzes and on paper problem solving activities.

- **Badges**: Individuals earn badges for various achievements which have little affect on student grades, but the teams with the most badges at the end of the semester win prizes.
- **Tests of Understanding**: We have 5 lower-stakes tests, rather than one or two high-stakes exams.

There are two primary limitations inherent in the approach to this teaching project. First, we lack a control group, therefore cause and effect are not able to be assessed. By tracking the DFW and retention rates over time, in ITIS 1212, 2300 and ITCS 1213 (programming II), we hypothesize that a distinct difference will be notable after intervention implementation. It should be noted that ITIS 1212 is a new course, so in years prior to 2013, we will observe cases of SIS majors who were enrolled in ITCS 1212, the CS version of intro programming. Second, we have implemented several teaching methods simultaneously, limiting the ability to discern which method was most effective. A qualitative post-hoc approach will be used to address this challenge.

D. Evaluation

Project evaluation will incorporate formative and summative evaluation comprised of a mixed-method design to guide modifications in the teaching approaches and to assess the degree to which these methods impact particular students for wider application of these approaches. Formative assessment will include collecting input from students enrolled in the reformed ITIS 1212 and 2300 courses, and discussing this information with the faculty instructors. Student assessment will include collecting the DFW and retention trends in introductory programming courses in the College to determine baseline rates prior to the teaching interventions compared to after the teaching interventions. End of course evaluations in the intervention courses will be used to measure impact of specific methods, along with follow up interviews with students to discuss the courses with the project evaluators. Summative assessment will include student learning outcomes, as identified in course outcomes rubrics, results from DFW and retention changes, student survey input, course evaluations, and interview themes. Because recruitment and retention data are lagging indicators, we will collect qualitative data from open-ended course evaluation and survey items, along with themes that emerge from all interviews conducted with student volunteers. Students who enroll in ITIS 1212 and 2300 will be invited to participate in interviews, with a goal of obtaining a cross-section of students who were successful and unsuccessful in the courses. Overall assessment of the project will be determined by evaluating activities and tracking participants throughout the grant lifecycle, presented in summative reports, as shown in table 1 below.

Table 1. Project Goals, Activities, Measures and Data Collection

Goals & Desired Outcomes	Activities	Example Measures	Data
Goal 1: Increase student performance in CS 1 and Web Applications courses Desired Outcome: Reduce occurrence of DFW grades	- Deploy active learning strategies in program introduction courses in SIS Dept (ITIS 1212 and 2300) such as flipped classroom	-Number/percentage of DFW grades in courses -Student course/module feedback -Student learning outcomes for courses	- DFW grade trends in courses since 2008 -Student course evaluations/module surveys -Student learning outcomes rubrics, self assessment surveys
Goal 2: Increase student performance in subsequent coursework Desired Outcome: Reduce occurrence of DFW grades and gateway course repeats	-Track student performance in ITCS 1213 (aka computer programming II) - Track student course repeats for ITCS/ITIS 1212/1213	- Number/percentage of DFW grades in ITCS 1213courses, disaggregated by feeder course(s) taken -Number/percentage of course repeats in ITIS 1212, 2300, and ITCS 1213	-DFW grade trends in courses since 2008

Goal 3: Increase student retention Desired Outcome: Increase Year 1 to Year 2 retention of SIS majors	-Deploy active learning strategies in program introduction courses in SIS Dept (ITIS 1212 and 2300) such as flipped classroom	-Number/percentage of students retained on a semester and annual basis	-Number/percentage of SIS students retained from first to second year, since 2008 -Number/percent of SIS students retained in semester following enrollment in new ITIS courses
Goal 4: Contribute to CS pedagogy research Desired Outcomes: Gather and disseminate evidence of specific intervention impact for various study populations	-Conduct mixed-method study of ITIS revised courses -Present findings to CCI, the University, and the CS Education community at large via reports, presentations, and proceedings	-Collect quantitative data and qualitative data -Number of presentations and proceedings	-DFW rates, student learning outcomes, retention rates; student interview themes -Presentations, reports, proceedings, and articles published

E. Knowledge Dissemination

We plan to disseminate our findings locally and nationally via the following vehicles. Locally, we plan to present findings and their formative implications to our College at a designated Faculty Meeting. We also plan to present our summative findings to the University community during the UNC Charlotte Teaching Week. At the national level, we plan to submit a conference proceeding paper to the Association of Computer Machinery's Special Interest Group on Computer Science Education (SIGCE) annual meeting in March 2015 that will be held in Las Vegas, Nevada.

F. Human Subjects

We have obtained Institutional Review Board authorization to conduct the student survey components of the study #13-04-03, "Impact of Flipped Classroom on Teaching and Learning". An amendment is underway for submission to the IRB office for review of the study in its updated plan. We anticipate approval and will conduct our study in compliance with University policy.

G. Extramural Funding

We are currently exploring extramural funding opportunities to support the expansion of pedagogical reform and to conduct larger scale study of these interventions in computing education. Funding opportunities include, but may not be limited to, proposal submissions to the National Science Foundation's STEM Talent Expansion Program (STEP) and the Transforming STEM Learning (TSL) program. Both of these programs provide support for research in computing education initiatives that seek to increase persistence and degree attainment. We believe that the SOTL funding support of this pilot study will greatly enhance our ability to compete for extramural funding.

H. Timeline

The overall project cycle is as presented below.

Table 2. Timeline

Timeline	Objectives and Outcomes	Responsible Person
Spring 2014	Identify SIS doctoral student to join research team	Maher
	Prepare GA for qualitative study	Rorrer, Bean
	Plan for project evaluation: amend IRB application, with survey & course evaluation updates, interview protocol	
	Interview currently enrolled students in ITIS 1212 and 2300; and previously enrolled students enrolled in ITCS 1212 and ITIS 2300	Rorrer, Bean, Graduate Student
Summer 2014	Gather DFW, retention rates for SIS overall, for ITIS 1212, 2300, and ITCS 1212, 1213	Rorrer, Graduate Student
	Create dataset including student performance, retention, survey, course evaluation data from 2008 to present	
	Analyze student course surveys; student learning outcomes to date	
	Review preliminary findings for formative adjustments	Maher, Latulipe, Long, Rorrer, Bean
Fall 2014	Ongoing project evaluation: conduct courses; continue measurements	Latulipe, Long, Rorrer
	Student interviews with past participants in ITIS 1212 and 2300 who retook course or left SIS major	Rorrer, Bean, Graduate Student
	Submit SIGCSE paper	Maher, Latulipe
Spring 2015	Summative reports and presentations to CCI, UNC Charlotte	Maher, Latulipe, Long, Rorrer, Bean

I. References

Bennedsen, and Caspersen, M. E. (2007). Failure rates in introductory programming, *SIGCSE Bulletin*, *39*, 2, 32-36.

Bevan, J., Werner, L., & McDowell, C. (2002). Guidelines for the use of pair programming in a freshman programming class. *Proceedings of the Conference on Software Engineering Education and Training* (CSEE&T'02), Covington, KY, Feb. 25-27, 2002.

Gehringer, E.F. and Peddycord,III, B.W. (2013). The inverted-lecture model: a case study in computer architecture. *Proceeding of the 44th ACM technical symposium on Computer science education* (New York, NY, USA, 2013), 489–494.

Hanks, B. (2006). Student attitudes toward pair programming. *Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education* (New York, NY, USA, 2006), 113–117.

Kaner, C. and Fiedler, R. (2005). Inside Out: A Computer Science Course Gets a Makeover. Available: http://www.kaner.com/pdfs/kanerfiedleraectprint.pdf

Mayer, R., Stull, A., Almeroth, K., Bimber, B., Chun, D., Bulger, M., Campbell, J., Knight, A. and Zhang, H. (2009). Using Technology-Based Methods to Foster Learning in Large Lecture Classes: Evidence for the Pedagogic Value of Clickers. Available: http//:www.nmsl.cs.ucsb.edu/papers/149.pdf

Office of Science and Technology Policy, (2012). "One Decade, One Million More STEM Graduates" Michael Feder, December 18, 2012. Available: http://www.whitehouse.gov/blog/2012/12/18/one-decade-one-million-more-stem-graduates

Porter, L., Guzdial, M., McDowell, C., and Simon, B. (2013). Success in introductory programming: What works? *Communications of the ACM, 56*, 8, (August 2013), 44-36.

The Pew Learning and Technology Program Newsletter, (2002). Vol. 4, No. 3 September 2002. http://www.thencat.org/PCR/PewNews/PLTP13.html

Roediger, H.L. and Karpicke, J.D. 2006. Test-enhanced learning: taking memory tests improves long-term retention. *Psychological science*. 17, 3 (Mar. 2006), 249–255.

Werner, L., Hanks, B., & McDowell, C. (2004). Pair-programming helps female computer science students. *Journal on Educational Resources in Computing (JERIC)- Special Issue on Gender-Based Computing Education, 4,*1, Article No. 3.