Building a Linked-Courses Learning Community for Introductory Development Majors

Amber Settle and Terry Steinbach
School of Computing, DePaul University, Chicago, Illinois, USA

Abstract – *Student recruitment and retention in computing, particularly of underrepresented groups, remains a significant issue for post-secondary institutions. Linked-courses learning communities, in which students simultaneously register for two related courses and participate in co-curricular and extracurricular activities associated with those courses, have been shown to have positive effects on students' sense of belonging, student-to-student and student-to-faculty interaction, and student retention. We describe a project which develops a linked-courses learning community for students in several development-focused majors. Details about course selection, student recruitment, and program evaluation are provided. The project serves as a starting place for other computing educators interested in leveraging the benefits of linked-courses learning communities for student recruitment and retention.*

Keywords: CS1, digital divide, linked-courses learning communities, retention, underrepresented groups

1 Introduction

According to the Bureau of Labor Statistics, employment of computer programmers is expected to increase eight percent from 2012 to 2022 [5]. The Computing Research Association reports the number of new undergraduate computing majors has steadily risen the last five years [27]. Many institutions have implemented multi-faceted efforts in retaining those majors through graduation which indicates there is still a problem with retention and persistence of these new majors [2]. Dorn and Tew [9] cite several STEM researchers who have found evidence of a relationship between student perceptions and learning outcomes. In particular, students' achievement expectations and self-concept were good predictors of success; students' self-assessment of their ability impacts their decision to major in computer science; student beliefs affect how they learn new information and student experiences both inside and outside of the classroom can shape their beliefs. Schutle and Knoblesdorf [20] found that computing experiences affect a student's perception and attitude which enable or inhibit pathways into the field. Factors such as limited faculty-student encouragement and involvement with other students in the major have been shown to have a higher impact on underrepresented groups in computing, such as women [8].

An approach that has shown promise in improving students' sense of belonging is linked-course learning communities [12]. The origin of learning communities dates back to a six-year experiment at the University of Wisconsin in 1927 [12], and numerous models have been developed. In a linked-course learning community, students simultaneously enroll in courses from different disciplines or interdisciplines that are connected in content, purpose, and organization [6]. Learning communities intentionally restructure the curriculum to connect students and faculty [12]. The community is designed to provide students with an integrative and collaborative learning environment, with the aim of enhancing student achievement, reducing attrition rates, and increasing student and faculty enthusiasm [6]. Other issues that learning communities attempt to address are inadequate levels of interactions among students and between students and faculty [11]. Another term for such experiences is first-year interest groups, although FIGs typically involve more than just two connected courses [12], [22].

While linked-course learning communities have a long history, there is relatively little to be found in the computing education literature about them. There are articles that discuss linking content between technical courses, for example, by connecting user-interface and software-development courses [18]. There are also researchers who argue that designing and delivering sets of multidisciplinary sets of introductory computing courses is beneficial both for computing majors and for others who take those classes [14]. Further, there are learning communities in existence at several institutions that have a focus on computing or other STEM disciplines. Rochester Institute of Technology has what it calls Living Learning Communities in which students with common interests can live together and spend a year exploring their topic area of interest with faculty and through organized activities [19]. Syracuse University has a Science, Technology, and Math Learning Community in which students live together and receive academic support including advising and tutoring [23]. And Purdue University has a Computer and Information Technology Learning Community where students take common courses and participate in organized activities together [17]. But there are no published articles to be found discussing the logistics of creating a technology-focused linked-courses learning community or evaluating the effectiveness or impact of such a learning community. This is surprising given that recent recommendations for learning communities suggest that they be intentionally engineered around combinations of courses

likely to be taken by students in clusters of majors [11]. Part of the problem may be that faculty who could create such communities lack administrative support or the ability to modify the courses to have integrative components as is recommended [22].

We focus here on the creation of a linked-courses learning community for computing majors. We describe a project currently underway at DePaul University to create a linked-courses community for two required first-term courses in three development-focused computing majors. We describe the courses chosen for the project and how they are being modified for the project. We also detail the process used to recruit students for the community, and outline the activities that will be a part of the co-curricular and extra-curricular components of the community. Finally we discuss the approach that will be taken to evaluate the impact of the linked-courses community on students' attitudes toward computing and on their study habits in programming courses.

2 Creating the Community

One of the first considerations when creating a linked-courses community is the target audience. As at many institutions, women and non-white men are underrepresented in the College of Computing and Digital Media (CDM) and particularly in the more technical majors found in the School of Computing. Our goal in creating this community is to improve the retention rates in these two populations. While the needs of the populations are different in some ways, they are placed together in the same community for practical reasons having to do with minimum class size. Efforts will be made to develop activities appropriate for both groups, and it is hoped that the mix of the two populations will produce some positive dynamics.

To ensure that students will be interested in joining the community two courses required in their major are the ones selected. It is also important to make sure that incoming freshmen are aware of the learning community and have sufficient opportunity to register for the linked courses. The creation of a linked-courses community is a challenging problem, in that students need to be aware of the community and what it can do for them when enrolling, which often occurs six months in advance of their first college courses. Reaching incoming freshman at that stage of the enrollment process involves coordination of efforts between staff at several offices in the university.

In the remainder of this section we describe the two courses chosen for the community, the approaches taken for student notification and recruitment, and the planned structure and activities associated with the linked courses.

2.1 Course choice and modification

The courses chosen for the project are required for each of the targeted majors to ensure that community is appealing for students and does not require undo effort to join. The majors targeted for this project are computer science (CS), computer game development: gameplay programming concentration (gaming: gameplay), computer game development: systems programming concentration (gaming: systems), and information assurance and security engineering (IASE). Each of these majors are ones in which students are expected to become strong programmers and have a significant number of courses focusing on application development in the curriculum.

Students declaring any of these majors have two choices for their first programming course. Students with little or no programming experience, which is a majority of students as expected from a recent nationwide study [10], are directed to take a Python programming sequence that makes no assumptions about prior experience. CSC 241: Introduction to Computer Science I and CSC 242: Introduction to Computer Science II focus on problem solving and application development in Python. The relatively simple syntax of Python allows the introduction of significant and interesting problems early in the course and provides the future developers targeted by the class with frequent opportunities for problem solving. Using Python to allow a focus on problem solving is something that other institutions have embraced [16]. The problem-solving approach in the class is enhanced by an extra 90-minute weekly lab section. The section is supervised by a graduate assistant, and during the session the students collaboratively solve programming exercises. The exercises are graded by the graduate assistant, and typically contribute 5-15% toward the course grade. Using a closed lab for programming classes has been shown to have positive benefits for retention, learning, and student satisfaction [3], [16].

Students who have earned a strong grade (above a B-) in at least one semester of a high-level programming language are directed to take a more accelerated Python course. CSC 243: Python for Programmers covers much of the material from CSC 241/242 but does so in a single quarter. The accelerated Python course focuses on a few applications that are particularly interesting when using Python, such as text processing, recursion, and HTML parsing, to ensure that students' procedural programming skills are strong. The class does not have a lab component associated with it. In-class activities include lecture and hands-on exercises, although the class is not held in a lab and not every student is required to bring a laptop.

Additionally, every freshman at DePaul is required to take a Chicago Quarter class. These classes are designed to acquaint first-year students with DePaul and the metropolitan community, neighborhoods, cultures, institutions, organizations, and people of Chicago. Each course has a Chicago-centric theme around which the academic topics are organized, and the courses come in two types: LSP 110: Discover Chicago or LSP 111: Explore Chicago. Discover Chicago courses meet for eight hours a day during the five-day week just prior to the start of classes, typically during the last week of August. The courses then have an abbreviated

schedule during the remainder of the quarter. Explore Chicago classes begin meeting when the regular quarter starts in early September, but meet for an extended class session. Each class is required to have multiple excursions into the city to see neighborhoods, institutions, and organizations relevant to the academic focus of the course. The course has standardized reading and writing requirements which are met using discipline-specific material designed by the instructor. The courses also have a "Common Hour," which addresses issues of transition for first-year students, including academic success skills and educational and career planning [7].

Students participating in the linked-courses community in development for this project will be enrolled during the autumn quarter 2014 in a reserved section of LSP 111 entitled "The Digital Divide." This Explore Chicago class explores the social issues surrounding access to information and communications technology (ICT). During the 1990s the digital divide was defined as access to the Internet and centered on racial and socio-economic differences. Today the divide has changed as technology has developed. Access has increased through the use of smartphones, but there are large differences in quality of connection, cost, and intent (entertainment versus empowerment). The course also considers the difficulties that people with disabilities face when trying to use ICT. A large focus of the course is the approach taken by the City of Chicago to address the metropolitan digital divide. The second co-author will be the lead instructor for the course, and she will be assisted by a student mentor from CDM.

Also during autumn 2014, the students participating in the linked-courses community will also be enrolled in a reserved section of CSC 241: Introduction to Computer Science I. The course normally has a prerequisite of college algebra, but this prerequisite will be waived for students in the community. The students will be given access to extra tutoring sessions and open hours to help them master the more mathematically challenging material in the course. If there are a sufficient number of students with programming experience in the linked-courses community, a second cohort of students in CSC 243: Python for Programmers will be created. The Python for Programmers class also requires college algebra which will be waived for community participants who will receive extra tutoring. Both the reserved sections of Introduction to Computer Science and Python for Programmers course will be taught by the first co-author. The lab section for Introduction to Computer Science will be taught by a graduate assistant who will be hired in June 2014. If possible that graduate assistant will also be hired for the extra tutoring sessions for the courses.

2.2 Student recruitment

The success of the linked-courses learning community will depend on recruiting sufficient numbers of students in the targeted population. With women and minority men representing a small portion of the overall enrollment in the CS, gaming: gameplay, gaming: systems, and IASE degrees, attracting as many as possible of the prospective students is crucial. In this respect the linked-courses community benefits from being implemented by faculty in CDM, a unit with a long-standing program for contacting prospective students as soon as they are admitted to the university [21]. Since 2004 prospective students have been paired with faculty members in the students' identified area of interest. Faculty members are directed to contact the students at least once and then to be available for questions from the students and their family. The program is unusual since undergraduate enrollment is centralized at DePaul and most units do not have contact with students until they agree to attend the university.

During the academic year 2013-2014, the authors have worked in conjunction with the CDM staff overseeing the prospective mentoring program. All students in the targeted population for the relevant majors and concentrations have been assigned to the authors as mentees. The students were initially contacted via email in early March 2014 with information about the linked-courses community. The authors also attended the in-person event for prospective students during March 2014. During the CDM Admitted Students' Day, prospective students and their families attend an afternoon-long information session and tour. The authors answered questions from their mentees about the community, CDM and DePaul as a whole. In April 2014 the authors reached out to students again when it became possible for admitted students to take their placement exams for DePaul University, taking the opportunity to remind them about the linked-course community.

Once the national May 1st acceptance deadline passed, students in the target audiences who identified one of the relevant majors or concentrations and submittd a deposit to DePaul were again contacted about joining the linked-courses community. Students who indicate interest will be given information about registration for the reserved sections of Explore Chicago and Introduction to Computer Science or Python for Programmers. Advising staff will assist them in registering for the appropriate courses before September 2014.

2.3 Planned activities

Although articles about computing-focused linked-courses communities cannot be found in the computing education literature, there are several learning communities in technology-focused programs in the U.S. including Purdue University [17], Syracuse University [23], and Rochester Institute of Technology [19]. These communities involve various co-curricular, extra-curricular, and communal-living opportunities. In developing the activities for the CDM linked-course community, the opportunities available to students in these programs were considered.

The CDM linked-courses community will involve at least one weekly co-curricular or extra-curricular activity. The activities will be organized and/or arranged by the authors, and as appropriate will involve other DePaul faculty and staff, staff

from local technology companies and organizations, and DePaul student organizations. The final schedule for the activities is pending, but the following describes the current plan for the quarter.

The co-curricular activities will include:

- Special lectures, for example, a guest speaker from a local game development company
- Community-service projects
- Visits to technology companies and/or organizations, primarily in conjunction with the Explore Chicago course

The extra-curricular activities will include:

- Meals with faculty mentors
- A video game night, possibly in conjunction with DeFrag, the CDM student gaming organization
- Field trips to technology organizations that do not fit into the academic topic for the Explore Chicago class, including a visit to the 1871 incubator, companies that have been started by DePaul alumni and students, and companies associated with the CDM advisory board
- Pairing of advanced undergraduate mentors with the students
- Organized activities in the newly formed Physical Computing Lab in CDM

3 Evaluation

An important component of the linked-courses learning community project is the evaluation of its effectiveness. Taking the first-quarter Python course alone may have an impact on student attitudes and study habits as well as on student retention, since it employs techniques that have shown to be effective in retaining and engaging students such as Python as the teaching language, closed labs, and small class sizes [3], [16]. It is important to measure what impact the linked-courses community has independently of the experiences of students taking the first-quarter Python course.

We have developed a survey instrument to measure the impact of the project. The survey of student attitudes and habits administered for this work was based on previous surveys of student attitudes, including the Computer Science Attitude Survey [26], the USG Student Computing Survey [25] used in the Georgia Computes! Project [4], and the Science Interest Survey [13]. Because the survey would be administered in person during sessions of the programming class, brevity was important. Questions were selectively chosen from previous surveys and modified to meet the needs of this study, producing a survey with 33 multiple-choice questions and a single question that asked students to rank resources used to obtain new computing skills. There were also several demographic questions to start the survey.

Each of the multiple-choice questions are listed below, and students are asked to answer the questions using a five-point Likert scale of 5 = strongly agree, 4 = agree, 3 = neutral, 2 = disagree, and 1 = strongly disagree.

1. I plan to major in a technology-related degree
2. I am sure that I can learn programming
3. I am sure I can do advanced work in computer science
4. I think I could handle more difficult programming problems
5. I can get good grades in computer science
6. I have a lot of self-confidence when it comes to programming
7. I'm not good at programming
8. For some reason even though I work hard at it, programming seems unusually hard for me
9. Computer science has been my worst subject
10. It would make me happy to be recognized as an excellent student in computer science
11. I'd be happy to get top grades in computer science
12. If I got good grades in computer science, I would try to hide it
13. I'll need programming for my future work
14. Knowing programming will help me earn a living
15. I will use programming in many ways throughout my life
16. Taking computer science courses is a waste of time
17. Once I start trying to work on a program, I find it hard to stop
18. I am challenged by programming problems I can't understand immediately
19. I am easily frustrated by difficult programming problems
20. I do as little work in computer science courses as possible
21. I like talking with my friends about programming
22. I like to program in my spare time
23. I belong in the computing field
24. I feel isolated in computer science courses
25. I am part of a community of programmers
26. Computer science offers good financial opportunities after graduation
27. Computer science allows me to be creative
28. Computing offers diverse and broad opportunities
29. I have a lot of support that will help me to succeed in computer science courses
30. Computer science provides opportunities to make a difference in the world
31. I have a lot of friends who are interested in computing
32. My family is happy that I am taking computer science courses
33. I have had good teachers in my computer science courses

There is one additional question that asks students: "Outside of your classroom studies, what are your resources for learning/obtaining new computing skills? Mark all that you

use and rank them in order of use. The one you use most should be ranked as 1, the one you use next often as 2, etc." The resources listed are: Friends/peers, Internet/web sites, professional organizations, self study, family members, tutors, faculty, and other (please specify). Students are instructed to leave a resource blank if they did not use it, so that only the resources utilized would be given a ranking.

This survey is in the process of being administered to every student taking the Introduction to Computer Science course during the 2013-2014 academic year, following procedures approved by the DePaul IRB. The results from this data will provide a baseline for understanding the impact that the first-quarter Python programming class has on student attitudes and habits. That information can then be compared to data gathered from students in the linked-courses learning community to gain a measure of the impact of the community on student attitudes and habits.

4 Discussion

The linked-courses learning community at DePaul addresses several important issues when trying to improve representation of underrepresented groups. In *Unlocking the Clubhouse*, Margolis and Fisher make the following recommendations for broadening accessibility to computing curricula [15]:

- Pay close attention to the quality of the student experience.
- Accommodate a wide range of computing experience among incoming students.
- Create a curriculum that reflects the many facets and impacts of computing.
- Establish structures for under-represented students to come together for support.

Prior to the creation of the linked-courses learning community, several of these recommendations were addressed in various ways. There are multiple paths for the introductory programming courses required by the CS, gaming: systems, gaming: gameplay, and IASE degrees, one for students with prior programming experience and one for students without any programming experience. Several student organizations exist to support underrepresented groups, such as HerCDM which is a mentoring and social group for female CDM students. There are a variety of degree programs at CDM, each with a slightly different focus on technology. In addition to the development-focused majors mentioned previously, there are degrees in information technology, interactive and social media, and more artistic areas like animation and digital cinema.

The linked-courses learning community builds on and extends these efforts, incorporating recommendations from the National Center for Women and Information Technology (NCWIT). NCWIT recommendations for improving female representation in computing [24] include:

- Recruiting strategically
- Retaining with student support
- Evaluating your efforts

The biggest contribution the community will make will be in enhancing the support structures for the development-focused majors. The students will share two classes, participate in co-curricular activities, and attend extra-curricular events as a group, which builds familiarity and a sense of belonging. It has been shown that student-to-student interaction is the most powerful predictor of students' intention to persist in the major beyond the first course [1], and the linked courses and required activities will promote this type of interaction. This is consistent with recommendations that student-to-student interaction be made a mainstream activity rather than an optional or purely extracurricular option [1].

Students will also be given extra tutoring and support to assist them in completing their first programming course, which should help build confidence and improve retention. Although the community is designed to last for a single quarter, students will have the option to enroll in sections together for follow-on classes, allowing them to create a cohort that lasts for several years. Formal evaluation of the program will be conducted to measure the impact of the learning community beyond the Introduction to Computer Science Python programming course.

5 Conclusion and Future Work

We present here information about the creation and evaluation of a technology-focused linked-courses community under development at DePaul University. The community links a general education course on the digital divide and a first-quarter programming class in Python, both of which are required in four development-focused majors in CDM. With the assistance of university staff, female and minority male students are being recruited from prospective students to join the community. Several co-curricular and extra-curricular activities are planned for the community, and students in the community will be provided with extra tutoring and support to improve their engagement and retention.

Evaluation of the implemented linked-courses learning community is important. The project will be evaluated using a survey instrument that is in the process of being given to every student taking the first-quarter Python course during the 2013-2014 academic year. Results from this baseline will be compared to the results from students in the community to understand the impact that the linked-courses community has on student attitudes and study habits. It is hoped that the results will show that the linked-courses community has a positive impact beyond what could be expected from taking

the programming courses alone, but that is simply a conjecture at this point.

Although linked-courses communities exist at other technology-focused institutions [17], [19], [23], there has been little attention paid to the development or evaluation of such communities in the computing education literature. This article makes headway into this area by providing a starting place for other computing educators interested in designing and implementing similar interventions.

Future efforts on this project include the analysis of the baseline information about student attitudes and habits in the Introduction to Computer Science course, implementation of the learning community, and a comparison of the data from the participants in the learning community and the other introductory Python programming students not in the community. Results from the evaluation of the project will inform modifications of the linked-courses learning community in future academic years.

6 References

[1] Barker, L.J., McDowell, C., and Kalahar, K. 2009. Exploring Factors that Influence Computer Science Introductory Course Students to Persist in the Major. In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (Chattanooga, Tennessee, USA, March 2009).

[2] Beaubouef, T. and Mason, J. 2005. Why the High Attrition Rate for Computer Science Students: Some Thoughts and Observations. *Inroads – the SIGCSE Bulletin* 37:2 (June 2005), 103 – 106.

[3] Boyer, K.E., Dwight, R. S., Miller, C.S., Raubenheimer, C.D., Stallmann, M.F., and Vouk, M.A. 2007. A Case for Smaller Class Size with Integrated Lab for Introductory Computer Science. In *Proceedings of the 38th ACM Technical Symposium on Computer Science Education* (Covington, Kentucky, USA, March 2007).

[4] Bruckman, A., Biggers, M., Ericson, B., McKlin, T., Dimond, J., DiSalvo, B., Hewner, M., Ni, L., and Yardi, S. 2009. "Georgia computes!": Improving the Computing Education Pipeline. In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (Chattanooga, Tennessee, USA, March 2009).

[5] Bureau of Labor Statistics, Occupational Outlook Handbook, Computer and Information Technology: Computer Programmers, http://www.bls.gov/ooh/computer-and-information-technology/computer-programmers.htm, accessed April 2014.

[6] Cargill, K. and Kalikoff, B. 2007. Linked Psychology and Writing Courses Across the Curriculum. *The Journal of General Education*, 56:2, pp. 83-92.

[7] Chicago Quarter, First-Year Program, DePaul University, http://liberalstudies.depaul.edu/FirstYearProgram/ChicagoQuarter/index.asp, accessed April 2014.

[8] Cohoon, J.M. and Aspray, W.F. 2006. *Women and information technology: Research on underrepresentation*. MIT Press: Cambridge, MA.

[9] Dorn, B. and Tew, A.E. 2013. Becoming Experts: Measuring Attitude Development in Introductory Computer Science. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education*. (Denver, Colorado, USA, March 2013).

[10] Ericson, B. and Guzdial, M. 2014. Measuring Demographics and Performance in Computer Science Education at a Nationwide Scale Using AP CS. In *Proceedings of the 45th ACM Technical Sympoisum on Computer Science Education*. (Atlanta, Georgia, USA, March 2014).

[11] Johnson, K.E. 2013. Learning Communities and the Completion Agenda. *Learning Communities Research and Practice*, 1:3.

[12] Kellogg, K. 1999. Learning Communities. ERIC Digest. Retrieved April 2014 from http://eric.ed.gov/?id=ED430512.

[13] Lamb, R.L, Annetta, L, Meldrum, J., and Vallett, D. 2012. Measuring Science Interest: Rasch Validation of the Science Interest Survey. *International Journal of Science and Mathematics Education.* 10:3 (June 2012), 643-668.

[14] LeBlanc, M.D., Armstrong, T., Gousie, M.B. 2010. Connecting Across Campus. In *Proceedings of the 41st ACM Technical Sympoisum on Computer Science Education*. (Milwaukee, Wisconsin, March 2010).

[15] Margolis, J. and Fisher, A. 2002. *Unlocking the Clubhouse: Women in Computing*. MIT Press.

[16] Newhall, T., Meeden, L., Danner, A., Soni, A., Ruiz, F., and Wicentowski, R. 2014. A Support Program for Introductory CS Courses that Improves Student Performance and Retains Students from Underrepresented Groups. In *Proceedings of the 45th ACM Technical Sympoisum on Computer Science Education*. (Atlanta, Georgia, USA, March 2014).

[17] Purdue University, Community and Information Technology Learning Community, http://www.purdue.edu/studentsuccess/orientation/learning_communities/profiles/technology/computer_information.html, accessed April 2014.

[18] Reimer, Y.J. and Cassens, M. 2014. Perspectives on Co-linking Design and Development Courses in CS. In *Proceedings of the 45th ACM Technical Sympoisum on Computer Science Education*. (Atlanta, Georgia, USA, March 2014).

[19] Rochester Institute of Technology, Living Learning Communities, http://www.rit.edu/academicaffairs/advising/learning-communities/living-learning-communities, accessed April 2014.

[20] Schulte, C. and Knobelsdorf, M. 2007. Attitudes towards Computer Science-Computing Experiences as a Starting Point and Barrier to Computer Science. In *Proceedings of the Third International Workshop on Computing Education Research* (Atlanta, Georgia, USA, September 2007).

[21] Settle, A., Pieczynski, S., Friedman, L., and Kizior, N. 2013. Evaluating a Prospective Student Mentoring Program. In *FECS 2013: The International Conference on Frontiers in Education: Computer Science and Computer Engineering* (Las Vegas, Nevada, July 2013).

[22] Stevenson, C.B., Duran, R.L., Barrett, K.A., and Colarulli, G.C. 2005. Fostering Faculty Collaboration in Learning Communities: A Developmental Approach. *Innovative Higher Education*, 30:1.

[23] Syracuse University, Science, Technology, and Math Learning Community, http://lc.syr.edu/future-students/which-one/communities/science-tech-math.html, accessed April 2014.

[24] Thompson, L.D., Eney, C., Davis, R., and Grady, T. 2014. Recruit and Retain Women in Undergraduate Computing: Success Stories using Research-Based Practices. In *Proceedings of the 45th ACM Technical Sympoisum on Computer Science Education*. (Atlanta, Georgia, USA, March 2014).

[25] USG Student Computing Survey, http://dl.dropboxusercontent.com/u/2635522/USG%20Student%20Survey.pdf, accessed April 2014.

[26] Wiebe, E.N., Williams, L, Yang, K. and Miller, C. 2003. Computer Science Attitude Survey, Dept. of Computer Science, North Carolina State University, TR-2003-1, http://www4.ncsu.edu/~wiebe/www/articles/prl-tr-2003-1.pdf.

[27] Zwben, S. 2013. Computing Degree and Enrollment Trends. http://cra.org/govaffairs/blog/wp-content/uploads/2013/03/CRA_Taulbee_CS_Degrees_and_Enrollment_2011-12.pdf, accessed April 2014.