

CyberCivics: A Novel Approach to Reaching K-12 Students with the Social Relevance of Computer Science

Jim Owens and Jeanna Matthews

Department of Computer Science

Clarkson University

8 Clarkson Avenue, MS 5815

Potsdam, NY 13699

{owensjp, jnm}@clarkson.edu

ABSTRACT

This paper describes CyberCivics, a novel approach to computer science outreach that integrates hands-on computing experiences with the study of contemporary social and political issues. We provide details of one such curriculum, focused on electronic voting, that we used with a high school AP Government course in Spring 2007. We describe our experience with this curriculum and how it enabled us to reach a larger, more diverse and more academically prepared group of students than prior outreach efforts targeted directly at computing courses. We also suggest a number of alternative approaches to outreach based on the CyberCivics idea.

Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]

General Terms

Education

Keywords

K-12, Outreach, Electronic Voting, Digital Rights Management, Security, Privacy

1. INTRODUCTION

A declining number of students in the United States are willing to study and earn degrees in computer science and related fields [9,15]. The number of students majoring in computer science has been in decline for decades, but the drop off has been particularly acute since the dot-com bust in the early part of this decade. In addition, the percentage of women choosing to pursue studies and careers in computer science is at an all-time low [16].

It has been widely acknowledged that many students shun computer science because they take a number of popular myths about computer science very seriously [3]. Some of these myths include:

- Few jobs exist in computing following the dot-com bust.
- The majority of existing computing jobs have moved offshore.
- Computing careers mean working alone writing code and not interacting with people or helping to solve important societal problems.

Outreach efforts often focus on AP Computer Science and other computing related courses in high schools; however, many students who are college bound and academically well-rounded do not take these courses because they have already rejected the idea of studying computer science. Thus, outreach efforts focused on elective technology courses can miss much of the diverse, well-prepared student audience we need to reach with experiences that challenge the damaging popular myths about computer science.

We can report first-hand experience with this problem. At Clarkson University, there is a well-funded outreach program targeted at grades K-12. Initially, despite some fairly aggressive marketing among local teachers, our K-12 outreach opportunities in computer science were limited almost exclusively to Technology classes, such as Principles of Engineering and Digital Electronics. In each case, these classes consisted of a small number (7-9) of exclusively male students. In addition, few of the young men in these classes had made plans to attend college after graduation, and in fact, most had not taken the math and science courses that would have prepared them to succeed as computer science majors. While these students obviously benefited from our outreach efforts, they did not represent a rich source of potential new computer science majors. From this experience, we concluded that what we needed to make computer science outreach more effective was relatively large classes of diverse students who were both college-bound and academically well-rounded.

During the spring 2007 semester, we tried a different approach to computer science outreach that we call "CyberCivics." We define CyberCivics as the study of contemporary social and political issues that grow out of computer science-related technologies. Examples include electronic voting, database privacy, privacy of electronic communications and transactions, intellectual property and digital rights management. Each of these topics lends itself to computer-based, technical solutions and allows us to incorporate hands-on activities in programming, network protocol analysis, database design and other core computing technologies.

We piloted CyberCivics in May and June of 2007 as a four-week unit on electronic voting in the AP Government class at a local high school, allowing us to reach a more diverse and high-achieving audience. Many of these students might never have considered taking a computing specific course on their own initiative. However, through the CyberCivics curriculum, we were able to reach them with activities that directly countered the damaging myths about computer science. We were also able to

reach the teacher with the important message that information about computing is crucial to the study of government.

In the remainder of this paper, we provide details of the electronic voting curriculum we used with the pilot class and describe some alternative CyberCivics outreach approaches. We end with a discussion of related work and conclusions.

2. INITIAL CYBERCIVICS EXPERIENCE

2.1 Background

As previously discussed, the impetus for the CyberCivics program grew out of our inability to reach a diverse, well-prepared student audience with computer science outreach activities. We made contact with an AP Government teacher in a local high school and presented her with a proposal, suggesting a number of possible topics involving both computing and contemporary social issues:

- Electronic voting
- Personal information privacy (medical, consumer, financial, etc.)
- Intellectual property and digital rights management
- RFID in retail products and "smart IDs" or E-passports, etc.
- Privacy and security of home and public networks

The AP Government teacher chose the electronic voting material as particularly relevant to high school seniors, many of whom were getting ready to vote for the first time. She offered us the four remaining weeks of class time following their AP Government exam in mid-May. In marked contrast with the classes we encountered in our earlier K-12 assignments, the AP Government class comprised 18 students, more than half of whom were female. In addition, most students in this class had plans to attend college immediately following graduation. In fact, this class was filled with the school's highest achieving seniors, including both the class valedictorian and salutatorian.

2.2 Four-Week Electronic Voting Curriculum

The electronic voting curriculum we developed consists of three units covering nearly four weeks of class time. The first unit introduces students to the basic ideas behind CyberCivics, with examples drawn from the full range of topics in our initial proposal, as well as some specific background information on direct recording electronic (DRE) voting systems. During the second unit, students learn the basics of programming in Python. Given this introduction to basic programming concepts and structures, students participate in an obviously "rigged" election in the third unit. Students perform a source code review of the electronic voting system software in an attempt to explain obviously flawed results.

2.2.1 Introduction to CyberCivics and e-Voting

The first unit in our pilot class opened with a discussion aimed at developing a definition for CyberCivics. Throughout the discussion, the instructor made frequent references to recent headlines on a wide range of related topics and issues, including electronic voting, privacy of electronic data and transactions, and digital rights management, among others (see Table 1). The point of this discussion was to make students aware of the myriad ways in which computer science-related technologies impact their lives on a daily basis. We illustrated that knowledge of computing was

necessary to be a well-informed citizen that could understand the issues behind these headlines.

Table 1. CyberCivics is the stuff of daily headlines

Veterans Administration Loses Data on 1.8 million

-- *Consumer Affairs*, 2/13/07

Sony BMG Hacking into CD Buyers' Computers

-- *FOX News*, 11/3/05

Audit Finds Many Faults in Cleveland's '06 Voting

-- *NY Times*, 4/20/07

Machines Record Votes Inaccurately in Tests

-- *Dayton Daily News*, 3/20/07

**Experts: Wi-Fi Eavesdropping Persists
Despite Stronger Security**

-- *Boston Globe*, 8/7/07

In the subsequent lessons of this initial unit, the focus narrowed to electronic voting. The class viewed and discussed the 2006 documentary film *Hacking Democracy* [1] that documents vulnerabilities in the electronic voting systems that count over 80% of the votes in US elections. This helped students to gain a better understanding of DRE voting machines as electronic devices that run potentially faulty computer software. The film also clearly illustrates that while software code reviews and other scientific/technical approaches are needed to address problems with DRE systems, technical solutions alone will not solve the whole problem. In another activity, students explored an interactive online demo of Diebold's (now Premier Election Systems) AccuVote TSX voting system [10]. The interface used in this touch-screen voting system served as a model for the Python-based voting system used in a mock election in the final unit of the course. [Unfortunately, as of August 26, 2007, all Diebold voting machine demos appear to have been removed from the Web.]

2.2.2 Learning to Program in Python

With DRE systems and the potential problems of their computer software as backdrop, the AP Government students learned the basics of programming in Python during the second unit. We selected Python for several reasons. Python is interpreted, rather than compiled, which can be easier for beginning programmers. Python syntax is also simple and straightforward, using indentation rather than braces and semi-colons to define code blocks. It is Open Source, with programming environments for a variety of platforms available at no cost. Since the local school had Windows laptops available for our use, we selected the Windows version of the standard Python environment for use in this curriculum. We also used Bryce Embry's excellent and freely available *Start Programming with Python* tutorial [7] to guide the programming activities in this unit.

In their first programming lesson, students entered a variety of statements into Python's interactive IDLE environment. Then they quickly moved on to writing small programs, such as those presented in the first seven lessons of the Start Programming tutorial. In most cases, we extended the programs in the tutorial in various ways. Students were also asked to write their own programs, based on simple exercises provided by the instructor.

Our goal for this part of the curriculum was not to turn the AP Government students into expert programmers. Instead, we sought to expose them to the workings of a high-level programming language and to provide them with a basic understanding of the fundamental programming structures. For the majority of these high school seniors, these lessons in Python represented their first experience with computer programming. With approximately two weeks' experience in writing small Python programs, the students were ready to put their newly-acquired skills to the test in a source code review of an obviously flawed DRE voting system.

2.2.3 Mock Election and Source Code Review

From the very first days of the electronic voting curriculum, the students were aware that they would participate in a mock election. With the AP Government teacher's cooperation, we added a real ballot question to the otherwise fictional election scenario: *Should AP Government students be required to take a final exam?* (In past years, there had been no cumulative exam after the AP Government exam was administered.) See Figure 1 below for a screen shot of how this ballot question appeared in the Python voting system.

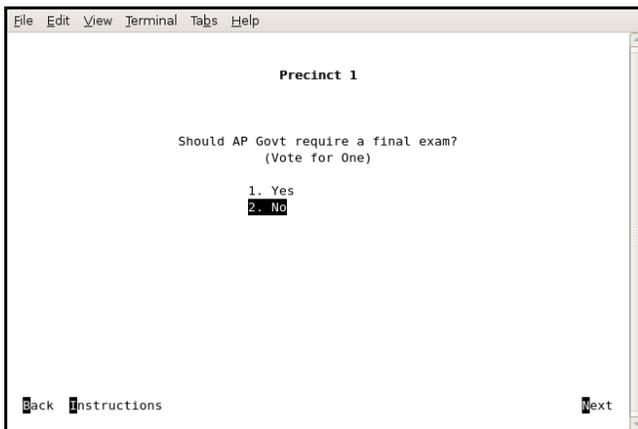


Figure 1. Ballot question shown in the Python voting system.

Every effort was made to make the mock election as realistic as possible. Several students volunteered to act as poll workers. Student poll workers organized the election, with the assistance of the instructor, who played the role of the voting system vendor/consultant. The voting system comprised a set of Python programs, provided by the instructor. Just prior to the balloting, the student poll workers set up and “zeroed” the election system. One-time access codes were then printed for all students to ensure that each eligible voter could cast only one vote. When preparations were complete, the students conducted and supervised the election, after which vote totals were generated.

Surprisingly, the final exam ballot question passed by an overwhelming margin! When the students protested, their teacher suggested that they obtain copies of the source code and review it. If problems with the voting system could be identified and explained, the teacher agreed to overturn the election results.

All students were provided print-outs of the Python source files for the mock voting system. In consideration of the students' relative lack of programming experience, the code was liberally commented and all identifiers were given meaningful names. Within the first class period following the mock election, a

student identified a function within the election source code that checked the results of the final exam ballot question and swapped the “Yes” and “No” vote totals in the event that “No” votes outnumbered “Yes” votes (see Figure 2). On the basis of this student's explanation, the AP Government teacher scrapped plans for the final exam.

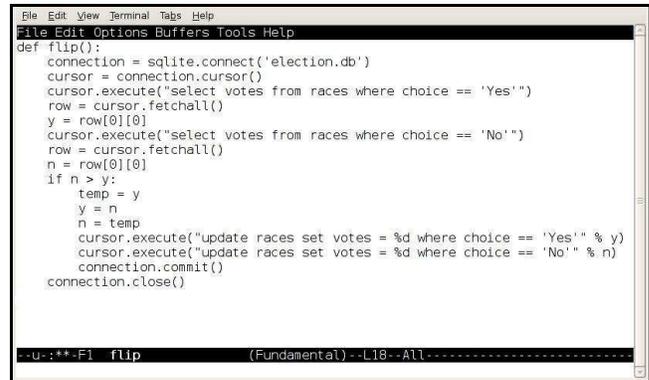


Figure 2. The flip function in the Python voting system conditionally reverses vote counts.

2.2.4 Culminating Activities

Following the mock election, we provided additional presentations on some additional CyberCivics topics, as well as an overview of computer science career opportunities, based on the online version of [17]. This activity gave us the opportunity to directly counter some of the damaging myths regarding the disappearance of computer science jobs. In fact, the Bureau of Labor Statistics (BLS) projects that job opportunities for computer scientists, database administrators, software engineers, and computer systems analysts will all grow “much faster than average” through 2014. In BLS terms, this projection translates to a growth rate of 27 percent or more. In addition, we presented some additional information on computer science as a career—salaries, working conditions, educational requirements, and more.

Later, the AP Government class made a field trip to Clarkson University's Applied CS Labs, where students participated in a number of demonstrations and hands-on activities related to computer and network security. These included a demonstration of cracking the WEP key of a wireless network, network traces of the students' own Web browsing and instant messaging activity, and sending an e-mail with a forged from header to the students' teacher, with her knowledge and permission. All these activities were designed to show students the importance of understanding the computing technologies they rely on in daily life.

3. BEYOND THE PILOT PROGRAM

The four-week electronic voting curriculum described above is just one of many possible applications of CyberCivics. Depending on the audience, the range of topics selected, and the time and facilities available, CyberCivics curriculum ideas could be developed to meet a variety of educational needs.

3.1 Alternative K-12 Outreach Ideas

This fall we will again be working with the same AP government teacher, delivering a two-week curriculum focused on the privacy of digital communications and databases. Whereas our spring

electronic voting curriculum was delivered after students had taken the AP Government exam, our fall curriculum begins early in the term and will be more tightly integrated with the existing AP Government curriculum. The CyberCivics activities will supplement and extend students' exploration of the U.S. Constitution, the Bill of Rights, and case law related to privacy and civil liberties. These activities will engage students in a variety of hands-on activities involving the privacy of unencrypted network communications, malicious software threats such as keylogging, data mining techniques, and similar exercises.

The successful lab visit component of the electronic voting curriculum, described in the Culminating Activities section above, provides a useful model for half-day field trips. Bringing the students to our own lab space allows us to design a wide variety of CyberCivics demonstrations and hands-on activities. These activities can be readily combined with short presentations on computer science career opportunities or other more traditional outreach activities designed to provide an overview of what computer science is all about.

We are also planning to integrate some CyberCivics curriculum into Project Challenge, another outreach activity aimed at high school students. Project Challenge is an annual program in which Clarkson University faculty offer short courses (five Saturdays for three hours each) to high school students. Clarkson's Computer Science Department has offered the course "Looking Under the Hood of Computer Systems" in Project Challenge for the past three years.

Beginning in the spring of 2008, we will also bring CyberCivics directly to K-12 teachers through the St. Lawrence County Science, Technology, Engineering, Mathematics (STEM) Partnership. Interested teachers from 19 school districts in our area will gain first-hand experience with CyberCivics concepts and activities in 35-hour hands-on workshops. Teachers will then be prepared to take CyberCivics curriculum directly to their own students and will also help to produce "gold" lesson plans for use by other teachers. Teacher involvement is key to reaching large numbers of students with the CyberCivics curriculum. Current funding will support these STEM workshops in CyberCivics through 2010.

3.2 Integration into CS1

This fall we are also experimenting with integrating CyberCivics topics and assignments into the first programming course at Clarkson. For example, students will write an electronic voting system for one assignment and will subsequently be asked to find flaws in a "solution" provided by the instructor.

CyberCivics in college computer science courses may not be as useful from an outreach perspective because, by definition, these students are already demonstrating an interest in studying computer science. However, these students will still benefit from the explicit ties to current social issues. This may help with retention of students, especially women.

3.3 Presentations for General Audiences

Finally, it must be recognized that the talented students we hope to reach through CyberCivics do not come by their misconceptions and mistaken beliefs about computer science in a vacuum. In our efforts to correct the misconceptions many have about our field, we must also reach out to parents, teachers, administrators, guidance counselors, and the whole community.

By their very nature, CyberCivics topics readily lend themselves to effective treatment in short, interesting presentations for general audiences. As noted earlier, many of the social and political issues related to the CyberCivics curriculum are reflected in daily news headlines. The computer science community has already taken a leadership role in addressing many of these issues at both the national and international level [18]. Fact-based presentations for school groups, service and other civic organizations could be most helpful in addressing these issues at the local level.

4. RELATED WORK

Numerous computer science programs and organizations are engaged in K-12 outreach and retention programs [4,6]. Some efforts focus on attracting and retaining women and students from other under-represented groups [2,12], often by offering an interdisciplinary approach [5,14]. Other efforts involve service learning [13]; these efforts can also serve to dispel the myth that computer science is asocial and unconcerned with real-world problems.

K-12 outreach activities include both in-school and "summer camp" programs. Many outreach programs include introductory programming activities related to gaming or robotics [11]. Web development activities are also popular. While many in-school programs involve partnerships with math and science teachers, in some states K-12 computer education is centered in the business department and outreach efforts are focused there [8].

The CyberCivics program we have developed is an in-school program intended to introduce diverse groups of academically well-prepared students to computer science in a way that helps to dispel many of the myths students may have heard about the field. We engage students in computer science activities and discussions that relate directly to real-world problems that are the stuff of daily headlines. Students' explorations of networking, programming, data mining, artificial intelligence, and related topics are tightly integrated into their social studies curriculum, which helps to reinforce their relevance. We are unaware of other attempts to integrate computer science material into high school social studies curriculum in a systematic way.

5. CONCLUSIONS

Based on our experience with the pilot class, it is clear that CyberCivics can be a powerful tool for computer science outreach. Our earlier in-school outreach efforts put us into small Technology classes of male students, many of whom had made no plans for college. In contrast, the AP Government class comprised 18 of the highest achieving students, more than half of whom were women. Absent their experience with CyberCivics, most of these students would have had no opportunity to explore computer science before graduating from high school. Few computing classes are offered in their school and, by their own admission, many students would not have selected such classes.

In addition to providing students with weeks of hands-on computing experience, we were able to challenge many of their misconceptions about computer science directly through presentations and discussions about the many exciting opportunities available to CS graduates. Students' exploration of electronic voting also helped to explode the myth that CS lacks relevance. Evidence to the contrary was reflected in the headline

news stories we discussed almost daily in class. No matter what their future career choices, it also armed them with information that can make them better citizens.

It also became clear to us that several students were bitten by the programming bug as a result of their CyberCivics experiences. During the second unit, when students were working through the programming exercises, several students began asking specific questions on Python's `random.choice()` function. Intrigued by the random text exercises in the programming tutorial, a group of students were working independently on a Python implementation of Rock, Paper, Scissors.

Perhaps the clearest evidence of the effectiveness of our outreach efforts came from a staff member whose son was a student in the AP Government class. She told us her son had registered midway through the spring semester for his first-term classes in the electrical engineering program he would attend in the fall. Given a choice between a scientific programming course and CS1, he had selected the scientific programming course. Following his CyberCivics experience, she told us her son contacted the university and changed his schedule in favor of CS1. We hope to be able to achieve similar outcomes in future CyberCivics offerings.

Finally, we believe that CyberCivics must "earn its keep" in AP Government or social studies classes if it is to remain viable. That is, high school teachers must see a positive return for the class time they invest in the CyberCivics curriculum. After all, these teachers are bound to be evaluated on the results their students achieve on the AP exam, or other standardized tests. We believe CyberCivics delivers in this area, as well. The high school teacher we worked with told us that she believes our curriculum helps to develop students' problem-solving skills and provides a rich source of relevant examples they can use in writing analytical essays. Overall, she agreed with us that CyberCivics helps educate better citizens of the modern world.

6. ACKNOWLEDGEMENTS

Many thanks to Diane Brouwer of the K-12 Project-Based Learning Partnership program for her help in making CyberCivics a reality. The GK-12 program at Clarkson University is supported by the National Science Foundation Division of Graduate Education (DGE-0338216). Thanks especially to April Martin and her students for working with us in the pilot class.

7. REFERENCES

- [1] Ardizzone, S. and Michaels, R. (Directors). 2006. *Hacking Democracy* [Motion picture]. United States: Teale-Edwards Productions.
- [2] Blum, L. and Cortina, T. CS4HS: An Outreach Program for High School CS Teachers. *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, pages 19-23, March 2007.
- [3] Cassel, L., McGettrick, A., Guzdial, M., and Roberts, E. The Current Crisis in Computing: What are the real issues? *Proceedings of the 38th SIGCSE technical symposium on Computer Science education*, pages 329-330, March 2007.
- [4] Cassel L., McGettrick, A., Davies, G., Topi, H. and Sloan, B. An Initiative to Attract Students to Computing. *Proceedings of the 38th SIGCSE technical symposium on Computer Science education*, pages 133-134, March 2007.
- [5] Condon, Anne. Connecting with Computer Science. *CSTA Voice*, September, 2007. Available at: http://www.csta.acm.org/Publications/Periodicals/CSTAVoice/csta_voice_09_2007.pdf.
- [6] Dodds, Z. and Karp, L. The Evolution of a Computational Outreach Program to Secondary School Students. *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education*, pages 448-452, March 2006.
- [7] Embry, B. 2002. Start Programming with Python. Available at: <http://www.bembry.org/technology/python/StartProgramming.pdf>
- [8] Ericson, B., Guzdial, M. and Biggers, M. Improving Secondary CS Education: Progress and Problems. *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, pages 298-301, March 2007.
- [9] Foster, A. Student Interest in Computer Science Plummet, *The Chronicle of Higher Education*, May 27, 2005. Available at: <http://chronicle.com/free/v51/i38/38a03101.htm>.
- [10] Premier Election Solutions. 2007. Products and Services. Available at: http://www.premierelections.com/products_services/voting/voting_equipment.htm.
- [11] Purdue University K-12 Outreach programs. Available at: http://www.cs.purdue.edu/external_relations/k-12_outreach/programs.
- [12] Rice University Computer Science Computing and Mentoring Partnership. Available at: <http://ceee.rice.edu/cs-camp/>.
- [13] Rosmaita, B. Making Service Learning Accessible to Computer Scientists. *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, pages 541-545, March 2007.
- [14] Sherrell, L., Liu, C., Pottenger, W. and Gross, P. NSF-DGE GK-12 Teaching Fellowships: Changing Student Perceptions about Computer Science. *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, pages 474-475, March 2007.
- [15] Tucci, L. College students continue to shun computer science, *CIO News: Headlines*, June 8, 2005. Available at: http://searchcio.techtarget.com/originalContent/0,289142,sid19_gci1096260,00.html.
- [16] Vegso, J. Interest in CS as a major drops among incoming freshmen. *Computing Research News* 17, 3 (May 2005). Available at <http://www.cra.org/CRN/articles/may05/vegso>.
- [17] U.S. Bureau of Labor Statistics. 2006. Occupational Outlook Handbook. Available at: <http://www.bls.gov/oco/oco1002.htm>.
- [18] United States Public Policy Committee of the Association for Computing Machinery (USACM), <http://usacm.acm.org>.