

Socially Relevant Computing

Michael Buckley
SUNY Buffalo
137 Bell Hall
Buffalo, NY 14260
1-716-645-3180

mikeb@cse.buffalo.edu

John Nordlinger
Microsoft Research
One Microsoft Way
Redmond, WA 98052
1-425-705-6211

johnnord@microsoft.com

Devika Subramanian
Rice University
6100 Main St MS 132
Houston, TX 77005
1-713-348-5661

devika@rice.edu

ABSTRACT

In this paper, we introduce socially relevant computing as a new way to reinvigorate interest in computer science. Socially relevant computing centers on the *use* of computation to solve problems that students are most passionate about. It draws on both the solipsistic and altruistic side of the current generation of students. It presents computer science as a cutting-edge technological discipline that empowers them to solve problems of personal interest (socially relevant with a “little s”), as well as problems that are important to society at large (socially relevant with a “capital s”). We believe that socially relevant computing offers a vision of computer science that has the potential to improve the quantity, quality and diversity of students in our discipline. We describe preliminary results from two on-going curricular experiments at SUNY Buffalo and at Rice University that implement our vision of socially relevant computing.

Categories and Subject Descriptors

K.3.2 [General topics]: curriculum issues, capstones.

Keywords

Socially relevant computing.

1. INTRODUCTION

In 2005, the U.S. graduated **54,588** bachelor degrees in Computer Science, but **86,031** in Psychology, **51,540** in Political Science, **53,391** in English and Literature, **80,545** in Arts and Music, and **73,389** in Communications and Library Science (source: caspar.nsf.gov). In 2006, the New York Times reported that more sports therapists graduated than engineers in the U.S. It is indeed ironic that at a time when computing has pervaded every aspect of our existence (email, iPods, cell phones, GPS, search engines, Facebook, eBay), and computation has become the major enabler of important discoveries in science (more than half of the breakthroughs of 2005 announced in *Science* would not have been possible without computation), we do not have a flood of students eager to study computer science.

We believe that part of the problem lies in the fact that we have not effectively used our curriculum to communicate the message that computer science offers a remarkable intellectual framework and an amazingly versatile set of tools to solve problems of great importance to them personally, as well as to society at large. All too frequently, undergraduate design courses lack social relevance

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGCSE '08, Month 1–2, 2008, Portland, Oregon, U.S.A.

Copyright 2008 ACM 1-58113-000-0/00/0004...\$5.00.

in both the little-s sense (how can computation help in organizing my files, my music, my vacation photos), as well as the capital-s sense (can we use computation to determine how to evacuate Houston in 72 hours under the threat of a Category 4 hurricane, to get rural farmers in India access to current market prices on cash crops, to make devices to aid people with memory, visual, or auditory impairments). We believe that computing for a cause, or computing as a means to solve problems, rather than as an end in and of itself, is key to attracting this generation to our discipline.

Research on instruction in Computer Science has shown the benefits on student comprehension and retention (and by extension, motivation) of design projects that reflect real-world experience [14], and that offer complicated and engaging challenges (e.g. [4]). Summary capstone design courses at the end of college are found to be critical in undergraduate preparation for careers in Computer Science [ABET 2000,3,12]. But true real-world problems have not traditionally been brought to undergraduate majors [6], and therein lies a missed opportunity to make Computer Science an attractive choice for incoming freshmen. While high levels of student engagement and intensity can frequently be found in courses carried out as one-time advanced courses and seminars (see[8]), it appears absent from freshman education, and is certainly not presented in much of the portrayal of Computer Science as an alternative to social science and the humanities. Exposing students to problems of social relevance throughout the curriculum offers the added benefit of addressing a common complaint that our students are not adequately prepared for the design challenges they face in their post-baccalaureate careers in industry.

We propose that by reconsidering Computer Science education in the light of societal and interpersonal relevance, and by coupling it with early outreach and improving the image of our profession, we can have a significant impact in increasing the number and diversity of students in our programs.

2. SOCIAL RELEVANCE

We suggest that the 4x rate of graduation in Social Sciences as compared to Computer Sciences is due at least in part because of students' desire to have a societal impact [2]. An anecdotal survey of freshman texts yields an inane collection of core examples: multi-chapter case studies on stick and coin games (*Java Java Java*, Morelli), a program to optimize pizza purchases (*Absolute C++*, Savitch), puppies and ducks as examples of integer length and class definition (*Head First Java*, Kathy Sierra). Most early programming courses are so focused on the mechanics of computing and details of programming language syntax (and our cause is certainly not helped by our fights over the “right” first

programming language) that our students fail to see what computation is for. This is akin to the way arithmetic was traditionally taught in K-12 in which students memorize the addition and multiplication tables rather than learn how to use multiplication and addition to solve actual problems. K-12 teachers have recognized this issue and devised curricula like the Chicago Math Curriculum which builds from the concrete to the abstract. Consider this concrete problem from the Web site of the National Council of Teachers of Mathematics (nctm.org):

Problem: The devil made a proposition to Daniel Webster. The devil proposed paying Daniel for services in the following way: "On the first day, I will pay you \$1,000 early in the morning. At the end of the day, you must pay me a commission of \$100. At the end of the day, we will both determine your next day's salary and my commission. I will double what you have at the end of the day, but you must double the amount that you pay me. Will you work for me for a month?"

Answer: Don't do it! Things start out well, but after 10 days, the salaries of Daniel and the devil come out equal, at \$51,200 apiece. On the 11th day, Daniel's salary doubles to \$102,400, but so does the commission he must pay the devil -- which means Daniel must fork over his entire day's salary. Two times zero equals zero, so on the 12th day Daniel doesn't get paid -- but the devil's salary doubles again! Now Daniel is out \$204,800, or more than he has earned the entire time. From there, things only get worse: By the 30th day, he not only has worked for free for most of the month, but has lost about \$1.9 trillion. The devil, meanwhile, has earned more than \$107 billion.

By explaining the relationship between this problem and credit card interest rates and sub-prime lending, students clearly see the relevance of the above puzzle to decision making in their everyday lives. In fact, close dovetailing of foundational material in computing along with actual applications is key. How should these applications be chosen? In the following sections we present several approaches ranging from problems of personal relevance to the 18-22 age demographic, to problems of broader scope that are important to the communities in which they live. For example, it is known (www.seejane.org) that the largest concern with teenage girls is their weight. Providing examples of the use of computation to design nutrition and weight management tools, as well as tools to gather information about health and well-being that go beyond weight loss, can engage young girls who may not otherwise consider careers in computing. There is a dearth of information regarding high schoolers and freshmen; their attitudes and their concerns. In [4], Chinese teenagers rated overpopulation and environmental pollution as their greatest concerns about the future; these were usually rated quite low by teenagers in other countries. Although still of concern to Chinese teenagers, nuclear war seemed more remote to them than it did to U.S. and former Soviet teenagers in earlier studies. We are exploring survey methods to map out concerns among potential freshmen in Houston and Buffalo in order to design meaningful examples for early programming curriculum. Our goal is to devise a set of modules akin to those in [11], which demonstrates how mathematics can be used to interpret and better understand every day events.

A socially relevant curriculum works not because it re-packages the same old concepts with spiffy, new, of-the-moment examples. As we show, it requires a new emphasis on problem representation and modeling, rethinking what the key concepts in computation are, rethinking the pedagogical order in which to cover them, and discarding concepts that are no longer central to the use of computation. The social computing curriculum is a departure from standard CS curricula, which focuses on core technical computing skills alone. The new curriculum teaches students how to learn about new domains, how to abstract and formulate problems using the tools and techniques of computer science, how to communicate and work in effective teams with scientists in other disciplines, as well as how to evaluate the social/ethical aspects of their solutions. It achieves a balance between foundational knowledge of computation, and the engineering know-how to implement solutions to problems of importance to society.

At the University of Buffalo and at Rice University, this effort has found its way into freshman courses, and in senior capstone design courses. We elaborate on them in the following sections.

3. THE SUNY BUFFALO EXPERIENCE

3.1 Introductory Programming

At SUNY Buffalo, we teach programming after we explore the following preparatory topics:

- Design and modeling
- How programmers view the world
- Problem space vs. solution space
- Programming with a larger methodology
- Labs and example problems that have a societal emphasis
- Eclecticism
- The Tao of Engineering

And our class examples have a societal relevance:

- Arrays: Model pollution drift through the Great Lakes.
- Random Numbers: Princeton Egg simulation – predicts the future, we consult it before continuing class.
- Tables: Build a drug interaction database.
- GUI: Design an augmentative communications device for the speech impaired.
- Control: a secure voting machine.
- Sorting: an mp3 jukebox.
- Exception Handling: Chernobyl simulation.
- We study the Therac-25 machine, the Hubble deformity, the Denver Airport baggage system, and other engineering errors.

This orientation of programmers as citizens culminates in a senior design project that has great social relevance and impact.

3.2 Capstone Course

CSE 442 *Software Engineering* is the first of a two-course capstone design sequence at the University at Buffalo. In this course students work in teams of 6-8 members, to design and implement software simulations of systems which solve complex and extensive real-world problems. The second course, CSE 453

Hardware / Software Integrated Systems Design, is a new addition to our curriculum in which students will take the software simulations developed by students in CSE 442 and realize them as hardware implementations and deliver the devices to the original clients.

Students expect capstone projects to be hard work and rewarding, but with the added component of socially relevance, they experience a targeted level of importance. They invent things to apply outside of their circle of friends and fellow students. The prospect of making a difference sustains and motivates them through the weeks and months of design, coding, and debug.

We have found that engaging students, faculty and the community in creating practical solutions to socially relevant problems focuses incredible philanthropic and creative energy. By working with the handicapped, students begin to see themselves as participating in a larger community, see themselves less as geeks and more as citizens. Students who were at academic risk or even considered switching majors continued in the program because they wanted to see these projects to completion.

3.2.1 An Augmentative Communications Device

This effort was based at a skilled nursing facility, Elderwood at Oakwood in Williamsville, NY, and involved “David”, a 43 year old stroke patient with limited motor skills, full cognitive capacity, but an inability to speak. David is in a wheelchair and has limited upper-body mobility. Students were asked to use commercial grade technology (laptop computers) to create a menu-driven means of selecting words and phrases that might enable David to communicate. While David was the initial client, the objective was to develop a product that could be used by others with similar disabilities. David visited our class, and students visited David in his residence. Despite the seeming complexity of the task, six teams produced a working “talker.” In the following semester students who wished to carry this project forward took the best of the class systems and assembled them into single system, added some additional features, and with the donation of a tablet PC from Microsoft Corporation, gave David a working device. He continues to use it as his main means of communication.



Figure 1: David using his new communications device to order pizza.

On the night he was given the prototype device by a student team, he used it to telephone one of the course instructors, and carried on a slow but discernible conversation. After hanging up, he used the device to order a pizza. He had not spoken on the phone in 20 years. In watching David it is clear how a little technology and a

lot of student energy went a long way to improving his quality of life. This was not lost on the students. Involvement with a student project team was a welcomed experience for David and his family, and staff at the facility where David lives participated with great enthusiasm.

3.2.2 An Augmentative Communications Device for Children

In the next offering students were asked to produce a child-centered version of the augmentative communications device, with graphics and auditory feedback. Devices for this population were expected to have a user-interface appropriate for non-readers, including menu items that address a child’s home, school, and medical settings.

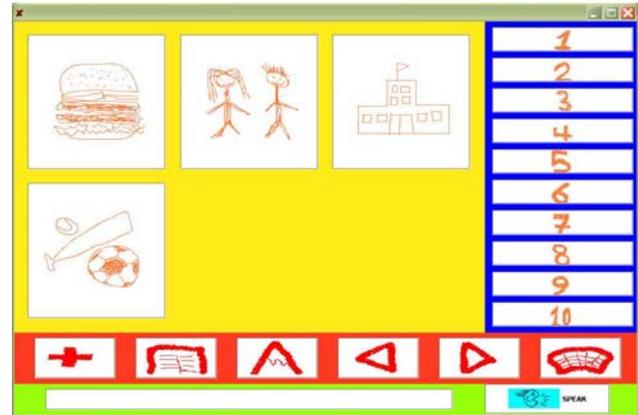


Figure 2: An alternative interface design for an augmentative communications device for a severely handicapped child.

The Center for Handicapped Children’s Learning Center (CHC) in Williamsville, NY kindly offered to serve as a specification and testing facility, including the use of its client base, faculty, therapists, nursing staff, administration, and equipment. Their participation ranged from simple non-intrusive observation of their students, to selecting a student to serve as a target client. The children at the Center are medically frail and technology-dependent. Their communications skills vary, but the school is a rich setting for adaptive technology. Young disabled children posed a unique set of problems for the instructors and students in the course. Student-developers had to consider auto scan and single switch menu selection, the varying reading skills of their student-clients, and varying gross and fine motor skills, and visual and audio impairment.

We thought that it might be necessary to address the emotional and psychological impact of severely handicapped children on our undergraduates. Not everyone was expected to visit the Center and we understood that engagement might be uncomfortable. In the end, no one refused, all participated, and everyone was changed by the experience, just as with David the semester before. Suddenly they were working on a project that could impact real families, and internalized the experience to ensure that the devices developed were truly useable. The child version of the augmentative talker is now in use by several children across the country.

3.2.3 The “DISCO” System

The objective of this effort is to enhance choice-making and cause-and-effect in physical, speech, and occupational therapy

sessions with physically and developmentally impaired children through the development of a programmable light and sound station. Utilizing light, music, and sound (including music and spoken-words), the station would help therapists and teachers create a choice-making, positive feedback or a calming environment for students who react positively to enhanced sensory experiences. The clients and customers are the teachers and therapists. The users are the handicapped students. The device kept statistics on successes and failures during use, so that teachers could alter the experience, and use it as part of a child's Individual Education Plan.

As with the Talkers, the students in the course immersed themselves in the client environment, visiting as often as was possible and asking teachers and therapists to audition prototypes. The enthusiasm of this undergraduate student teams for helping handicapped children through the use of computers, software, stage lighting, fog machines, fiber optics, water, sound, music and video had an interesting and discernible artifact: the college students felt that they were bringing their culture into new territory. One student said "this is what I do, this is what I know, and now I can use it to help someone". Some students purchased equipment on their own to build prototype systems. Students have dubbed it the DISCO (Disabled Interactive Sensory Coordination Opportunities) system. In all, 17 teams designed incredible software solutions, which varied significantly and demonstrated very divergent but creative approaches.



Figure 3: A live demonstration of the DISCO system.

4. THE RICE UNIVERSITY EXPERIENCE

In the Fall of 2007, we are offering a new capstone course on hurricane risk assessment and the design of evacuation policies for the city of Houston. The course is team taught by Devika Subramanian from computer science, Bob Stein from political science and Leonardo Duenas-Osorio from Civil and Environmental Engineering. The course has attracted a wide base of students from both engineering and social sciences, ranging in experience from freshmen to seniors.

The major driver for the course is the Office of Emergency Management in Houston which is interested in the development of computational tools to design and evaluate evacuation and sheltering policies in the event of a major hurricane. The Houston metropolitan area is home to over 5.5 million inhabitants; it is the

sixth-largest in the United States. When Hurricane Rita bore down on Houston, 2.5 million Houston area residents were evacuated, making it the largest evacuation in the history of the United States. Critics of the evacuation process believe that authorities waited far too long [10] to permit outward-bound cars to use both sides of the Interstate highways. It is estimated that residents who decided to leave Houston (and that did not need to evacuate) had a significant impact in the mobility of populations at true risk. Indeed, if a mere 15% of Harris county residents had sheltered in place in their homes, the transportation system could have absorbed evacuees from the more vulnerable Galveston area. Instead, thousands of residents clogged major freeways and witnessed gasoline shortages, and difficult recovery. The need for computational tools which can help regional authorities develop sound evacuation and sheltering policies has never been more critical.

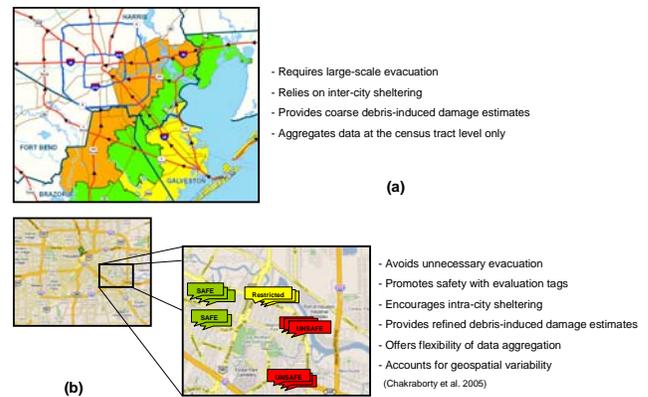


Figure 4. (a) census tract level damage estimation enabling zip code evacuation, and (b) household-level risk assessment enabling informed implementation of mitigation actions and individual evacuation decisions.

To solve this problem, we have put together inter-disciplinary teams of undergraduate students in computer science, political science and civil engineering. The city has provided us a large database containing information on the structural condition of all residential and commercial properties in the area. The first sub-task for the teams of undergraduate students in the class is to mine the database to assess household-level flood and wind damage risk and to identify regions of the city that are particularly vulnerable. Figure 4c above shows every property at a street by street level tagged according to their hurricane-worthiness by this data mining computation. Teams of civil engineering and computer science majors are currently engaged in building a computational tool that tags properties and displays them using state-of-the-art GIS mapping tools. To the freshmen on these teams, many of whom have AP credits in computer science, this is a major exercise in computational modeling of structural risk assessment, and learning how to work with civil engineers, GIS experts and CS upper classmen to construct a software engineering solution to a real-world problem. To many engineering freshmen, this will serve as a first exposure to the use

of computing in a very important context. The second sub-task is to use the calculated information on safety of properties to estimate the number of people who need to evacuate and their geographic distribution. Working with political science students, they will conduct surveys of the population to determine attitudes to evacuation, e.g., likelihood of evacuation, likely destinations, etc to construct transportation loading curves for use with commercially available transportation simulation systems. These will seed the third part of the course --- constructing household-level agent-based models [1] of traffic to observe the impact of specific evacuation/sheltering decisions made by the City Government. The tools built by the students will be demonstrated before City officials in December 2007 for potential inclusion in planning for future disasters. The course will be formally assessed by the Center for Civic Engagement at Rice University. A more detailed report of our experiences with this course will be produced at the end of the year, and will be available in early 2008.

5. CONCLUSIONS

We propose socially relevant computing as a new approach to constructing a computer science curriculum. Socially relevant computing emphasizes the use of computation for solving problems of personal and societal interest to students. It offers opportunities to demonstrate that computer science is a mainstream endeavor and that it offers conceptual and technological tools for solving meaningful, real-world problems. Courses in this new framework help students identify and model tasks, and design and implement computational solutions that show deep understanding of their embedding in the world. At the very least, socially relevant computing offers interesting examples to illustrate foundational concepts in computer science. By emphasizing problem-solving, and by giving students practice in recognizing needs and engineering solutions to them via computation, socially relevant computing at its finest promises to create a more entrepreneurial, as well as a more broadly educated computer scientist.

In a nutshell, we propose positioning computer science to teenagers this way: you can make the world a better place (“Socially relevant”), have more fun and be cooler while doing it (“socially relevant”), and be technical marvels with the latest software and applications. Why wouldn’t computer science be the best major on campus then?

6. ACKNOWLEDGMENTS

Devika Subramanian thanks the City of Houston and the Center for Civic Engagement at Rice University for their support of the course on hurricane evacuation. John Nordlinger thanks Microsoft Researchers Jim Gray, P. Anandan and Sailesh Chutani for their encouragement, guidance and thoughtful conversations in this area.

7. REFERENCES

[1] Bonabeau, E. 2002. Agent-based modeling: methods and techniques for simulating human systems. *Proceedings of the National Academy of Science*, 99(3):7283-7287.

[2] Buckley, M., Schindler, K., Kershner, H., Alphonse, C., and Braswell, J. 2004. Benefits of Using Socially-Relevant Projects in Computer Science and Engineering Education,

Proceedings of the 2004 American Society for Engineering Education.

[3] Clear, T., Goldweber, M., Young, F.H., Leidig, P.M. and Scott, K. 2001. Resources for Instructors of Capstone Courses in Computing, *ACM SIGCSE Bulletin*, 33:4, December 2001.

[4] Dannelly, R.S. and Steidley, C.W. 2001. A Student Laboratory Environment for Real-time Software Systems Development, *The Journal of Computing in Small Colleges*, ACM, 16:3, March 2001.

[5] Dodds, J. and Lin, C. 1992. Chinese Teenagers' Concerns about the Future: A Cross-National Comparison, *Adolescence*, 27:106, pp 481-86 Summer 1992.

[6] Gehrke, M., Giese, H., Nickel, U.A., Niere, J., Tichy, M., Wadsack, J. and Zündorf, A. 2002. Reporting about Industrial Strength Software Engineering Courses for Undergraduates, *Proceedings of the 24th International Conference on Software Engineering*, ACM, May 2002.

[7] Hayes, J.H. 2002. Energizing Software Engineering Education through Real-World Projects as Experimental Studies, *Proceedings of the 15th Conference on Software Engineering Education and Training*, IEEE, 2002.

[8] Howerton, C.P. 1998. Cactus Systems: a Computer Science Practicum that is more than a Capstone, *ACM, Proceedings of the 19th SIGCSE Technical Symposium on Computer Science Education*, 20:1, February 1998.

[9] Jorgensen, J.E., Mescher, A.M. and Fridley, J.L. 2001. Industry Collaborative Capstone Design Projects, *Proceedings of the International Conference on Engineering Education*, 2001.

[10] NPR (2005). “Q&A with Joanne Nigg on U.S. Evacuation Plans, Post-Katrina”. September 30, 2005. <http://www.npr.org/templates/story/story.php?storyId=4865033>

[11] Paulos, J.A. 1996. *A mathematician reads the newspaper*, Anchor Books, Doubleday, 1996.

[12] Rover, D.T. and Fisher, P.D. 1997. Cross-Functional Teaming in a Capstone Engineering Design Course, *ASEE/IEEE Frontiers in Education Conference*, 1997.

[13] Ruud, C.O and Deleveaux, V. J. 1997. Developing and Conducting an Industry Based Capstone Design Course, *Proceedings of the 27th Annual Frontiers in Education Conference (FIE 1997)*, 2:644-647, 1997.

[14] Sikkil, K., Spil, T.A.M and van de Weg, R.L.W. 1999. Replacing a Hospital Information System: An Example of a Real-World Case Study, *Proceedings of the 12th Conference on Software Engineering Education and Training*, IEEE, 1999.