Paper ID #6376

Introducing 6-12 Grade Teachers and Students to Computational Thinking

Dr. A. Dean Fontenot, Texas Tech University

Dr. Fontenot is the Sr. Director of the Texas Tech T-STEM which provides professional development for K-12 teachers as part of the Texas STEM (T-STEM) initiative. The Texas Tech T-STEM Center focuses on project-based learning with the integration of the engineering design process. The Center provides professional development training for T-STEM Academies, T-STEM Early College High Schools, and all Texas school districts, public and private. She collaborates with Whitacre College of Engineering Faculty as well as faculty from other colleges and universities on STEM related grants, activities, and endeavors. Dr. Fontenot teachers Professional Communications for Engineers: practical applications to written, oral, and Internet communication as well as an introduction to engineering ethics and service learning. She sits on numerous councils and boards, including the Texas Alliance for Minorities in Engineering Board, Texas Tech Teacher Education Council, the Pre College Engineering Council, and the Service Learning Faculty Advisory Council. She publishes in engineering related journals and presents at STEM conferences like ASEE, STEM Best Practices, and Frontiers in Education.

Mr. Richard A Burgess, Texas Tech University

Richard Burgess currently works as the Associate Director of the Murdough Center for Engineering Professionalism (MCEP) and National Institute for Engineering Ethics (NIEE) at Texas Tech University. He oversees the day to day operations of the Center's distance learning courses for both engineering students and practicing engineers. Additionally, he co-teaches an on-campus ethics course for undergraduate students. Burgess provides guest lectures on ethics throughout the Whitacre College of Engineering. Burgess has also begun working closely with Texas Tech's T-STEM Center to incorporate ethics into K-12 STEM education. The push to increase the number of students pursuing STEM careers needs to be accompanied by a sophisticated understanding of the complexity of technology. Ethics is a key part of this complexity and the next generation of STEM professionals will need the skills to effectively engage the ethical challenges they will face. Burgess is a regular presenter on incorporating ethics in a K-12 setting. A theme throughout these roles is the importance of teaching ethics and promoting ethical reflection in a way that is both accessible and substantive. This is a challenge that Burgess is keenly interested in. He holds bachelor's and master's degrees in Philosophy.

Ms. Vinitha Hannah Subburaj M.S
Ms. Debra J Nash, Texas Tech University T-STEM Center

Debra Nash is the Associate Director of the Texas Tech University T-STEM Center. She works with middle and high school teachers and administrators to help them implement science, technology, engineering, and math (STEM) curriculum. Nash also helps them learn to use project-based learning. In addition, she works with the Texas STEM Coalition, and with the Texas Tech T-STEM Center's social media sites.

Introducing 6-12 Grade Teachers and Students to Computational Thinking

Abstract

Computing disciplines struggle to increase student retention rates. Creating interesting curriculum to attract freshmen and help them learn is challenging. Computational thinking has become a key concept to motivate students with problem-solving strategies and has been used across various fields irrespective of computing majors. This paper describes the vertical integration of middle school and high school students with undergraduate students at a state university in a computational thinking project. Middle school students adopted the role of customers/end-users developing project requirements, high school students were designers, and undergraduate students were developers actually implementing the software system. One of the main themes of this project was the importance of ethics in computational thinking. To this end, teachers were provided with information and coaching on how to teach ethics. Students were provided with introductory ethics discussions and given opportunities to identify and reflect upon the ethical dimensions of computational thinking in the context of several of the aforementioned activities. This paper will briefly highlight and evaluate the ethics pedagogy adopted in this project. The results obtained by carrying out the computational thinking activities among a diversified group of students with varying knowledge and expertise levels were promising. The vertical integration had challenges and research components, which, when unveiled, will attract new students to computing disciplines.

Keyword: retention rate, computational thinking, vertical integration, ethics

Introduction

An NSF funded project CPATH-2: Collaborative Research from Middle School to Industry: Vertical Integration to Inspire Interest in Computational Thinking (CPATH) was started in 2008. The main objective of the project was to achieve vertical integration and introduction of computational thinking among K-12 middle school and high school students and college undergraduate students to increase intellectual thinking and increase the retention rate in computer science departments. The CPATH project focuses on common computational thinking interests and affinities. The principle goal of the project was to achieve vertical integration and transition of secondary students and teachers into undergraduate programs focused on computational thinking. The project goal was to develop an iPad/iPhone application to simulate the environment inside the NASA Orion Mock-up Space Capsule, and to automate the tasks manually performed by humans inside the space-craft. This project goal was achieved by introducing computational thinking activities to middle school, high school, and undergraduate students.

Associated with the field of Computer Science, computational thinking can be described as a modern way of thinking and problem solving. With emerging science and technology, computational thinking helps one view a problem in terms of the nature of the computation involved rather than just programming and coding. Wing [1] describes computational thinking as a fundamental skill that is needed for everyone. Also in the paper, Wing states that computational thinking is a recursive and parallel process. Solving a problem using computational thinking involves solving a larger complex problem by breaking it into smaller parts. In this project, the problem of developing an iPad application (small-screen application) was solved by performing smaller computational thinking activities.

Computing deals with the study of computers and the way they are programmed. Computer science emphasizes computational concepts and the problems associated with them. Like other science disciples, computing is also an important discipline that teaches basic principles and ideas rather than computer programs. In the project that was carried out, we emphasized computing concepts to add critical thinking and ethical dimensions to the computational thinking activities. The computing skills are fundamental and would teach the students the fundamentals of computers but would not focus on just programming. The frustrating misconceptions about computers can be resolved when the principles of computing are understood clearly. The activities that were developed for the middle school and high school students had the underlying factor of taking computing concepts to students in an exciting manner.

Role of ethics in this project

As emphasis on K-12 STEM education increases, there is a growing appreciation for the importance of ethics in education and outreach efforts. However, this is still an inchoate trend in STEM and thoughts about best practices in this context are still being developed. It is worthwhile, then, to explore how ethics might be effectively incorporated into STEM education. Specifically, this section will review the effort to integrate ethics into the CPATH project.

One of the major goals of the CPATH project is to instill in students and teachers the idea that computational thinking is more than a heuristic for solving technical problems. Computational thinking takes place in a social context which includes an ethical dimension. Stated more directly, ethics is an integral part of computer science and engineering practice. There was a conscious effort to integrate ethics alongside the more technically oriented content rather than treat it as a discrete after thought. In fact, some of the ethics content integrated in the CPATH curriculum was not explicitly identified as “ethics”. Rather, in some instances, questions about the psychological impact or the cost-benefit dimensions of a given choice were considered in order to get at some of the core elements of ethical deliberation. This integrated approach is intended to convey that social and ethical considerations are as important as the technical dimensions of a given design or project. Moreover, efforts were made to sensitize students to some of the subtler, more complex issues raised by computer technology. Engineering and computer ethics educational efforts are sometimes dominated by the use of oversimplified case studies wherein both infractions and solutions are obvious. Yet, such case studies often fail to model the complexity of the day to day application of technology. It was important, then, to help the students see and begin to grapple with this complexity. This strategy has, in the authors’ opinion, a secondary benefit which will be discussed at the conclusion of this section.

While the overarching strategy to teaching ethics was an integrated one, some preparatory work was necessary with both the teachers and students at a rural school district in west Texas. The teachers were to be the primary facilitators for ethics discussion and reflection (with support/expertise being lent by engineering ethicists at a major university). To this end, two middle school and high school teachers and the school superintendent were brought to the university for, among other things, a mini-workshop on teaching ethics in STEM. The workshop began with a survey of some of the characteristics (e.g. complexity, conflicting interests, and finite resources) of computer science and engineering practice. Discussing these characteristics makes for a straightforward transition to talking about ethics. This is important insofar as there is sometimes a tendency to think of computer science as being removed from the social context it takes place in and thus focused exclusively on programming and hardware details. Talking about complexity, for example, allowed for a natural segue into the ethical challenges computer scientists and engineers face when applying even simple sounding prescriptions like upholding "the safety, health, and welfare of the public" [2]. Defining concepts such as 'safety' and 'welfare' in practical, actionable terms can be difficult. Reconciling tensions between the interests of the public and those of a client can be deceptively complex. It was important to establish these points with the participants in order to both motivate the inclusion of ethics in the CPATH curriculum and communicate a nuanced view of engineering and computer ethics.

After this initial groundwork was laid, time was spent defining what ethics is as a field of study. Though many people talk about ethics in their daily lives, understanding about what it is to "do ethics" can be highly variable. Some basic methodological points were made in the workshop (e.g. emphasis on the Socratic Method) along with distinctions necessary to teach ethics effectively. Among these distinctions was the difference between the expression of a principle and its origin (e.g. the difference between the expression of the Golden Rule and where it draws its origin/justification from). This distinction is important to keep in mind when teaching ethics insofar as some teachers may feel like, in talking about ethics in the classroom, they are infringing upon parental territory. However, when teaching ethics in a K-12 setting, teachers should be focusing on the expression and application of principles and not on where those principles came from (e.g. the Bible, the Quran, or the Confucian Analects). These distinctions transitioned nicely into a discussion of potential sources of ethical guidance such as the codes of ethics for the Association for Computing Machinery [3] and National Society of Professional Engineers [2]. Finally, the workshop included a discussion of several challenges associated with teaching ethics. These challenges were broken down into two categories. The first category addressed objections common in ethics discussions regardless of grade level. For example, common "ism's" such as relativism and absolutism were described and addressed. The second category highlighted potential difficulties with including ethics in a middle and high school setting (such as the one discussed above). This workshop covered foundational concepts necessary to effectively teach ethics. However, it was not without its shortcomings. One of these will be discussed shortly.

In addition to the workshop with the middle and high-school staff, preparatory time was spent with the middle and high school students who would be participating in the CPATH project. These students visited Texas Tech and were given a brief introduction to ethics (roughly 45 minutes). The introduction began with the students watching a trailer for the documentary Windfall (the trailer can be viewed on YouTube at http://www.youtube.com/watch?v=8OZgoERceSU). While this trailer clearly communicates a fairly negative view of wind energy, it was useful to introduce a discussion of the consequences (intended and otherwise) of technology. From here, students were given an introduction to ethics in the context of the CPATH project. They were asked to consider the ethical dimension of the work they were doing. Examples from the CPATH activities were given (e.g. issues in data collection and the reporting and privacy issues that arise in binary coding) to help prepare the students to reflect on the social and ethical impacts of computational thinking. Finally, students were given several steps to follow when analyzing ethical issues. In hindsight, these steps should have been reinforced with a handout for both the teachers and students.

Once this preparatory stage was completed, teachers and students were free to begin the CPATH activities geared towards teaching computational thinking. To reiterate, the approach to teaching ethics was an integrated one. Most of the major activities included an ethics component though it was not always labeled as such. The teachers were the primary facilitators of ethical reflection through the use of open-ended questions, group discussion, and journaling. This effort was reinforced by the CPATH team members through site visits, video meetings, and e-mail communication. The educational objectives (with respect to ethics) were fairly modest. It was hoped that students would walk away with an enhanced sensitivity to and appreciation for the ethical dimensions of computational thinking. Additionally, it was hoped the students would begin sharpening their ethical reasoning. Several of the activities utilized in the project are included in the appendices of this paper. Though the overall approach to the project was to integrate ethics in other activities, one exercise was developed with an explicit emphasis on ethics content. It is included in Appendix B.

As of the completion of this paper, final quantitative assessments of the CPATH project are still underway. However, even when this data is collected it remains the case that success in ethics education is notoriously difficult to quantify. Aspects that can be straightforwardly measured are not always substantive in nature. Success, therefore, is often appraised at a qualitative level. In this regard, early indications are that the project was, on balance, successful. Several articulate students nicely summed up what they learned about the importance of ethics in a presentation to CPATH staff and School Board members. The student comments were developed with very little teacher guidance; lending additional credibility to the idea that this intervention was successful. The approach to integrating ethics in CPATH was by no means exclusive to computational thinking. It is hoped that both the students and teachers will find ways to include and discuss ethics in other content areas. This is key to cultivating effective STEM professionals and an informed citizenry.

It is easy to identify shortfalls to any project in retrospect and this one is no exception. Despite the preparatory work, the ethics component of the CPATH project got off to a slow start. The school teachers and students alike were unsure about how to discuss ethics in the context of the various CPATH activities. There was some confusion about what the ethics questions were asking and this reflected a more fundamental uncertainty about what ethics really amounts to in a STEM context. This is not to say that the teachers and students are not familiar with ethics in general. Rather, the unfamiliarity is a function of the relatively novel emphasis on ethics education in the classroom. For reasons already alluded to in the beginning of this section, this is not surprising.

In the context of the project, this slow start and confusion can be largely attributed to the approach that the CPATH ethics team members adopted. While laying the groundwork for effective ethics education is important, a more innovative approach would have led to quicker results. The details of such an approach are still being formulated. Prima facie, however, it seems clear that more time modeling ethics education was necessary. While it was important to define ethics and discuss methodology, much more time should have been spent “doing ethics” with the teachers. The dialectic nature of ethics education can make it difficult to teach in succinct terms. Effective teaching really stems from experience in this regard. Some of the information covered in the mini-workshop could have been conveyed through reading material. This would have left time for modeling ethics discussions and activities that would have, likely, led to the teachers feeling more comfortable incorporating this content into the classroom.

It is the hope of the CPATH team that both the teachers and students at the school developed a fuller appreciation for the ethical complexity of STEM disciplines and the obligations professionals have to the public. There is reason to believe the project was, as whole, successful in this regard. This would be reason enough to justify the ongoing inclusion of ethics in STEM education. However, there is another potential benefit to ethics education that provides substantial justification for its inclusion. Talking about ethics in STEM emphasizes the complex but very interesting challenges professionals in these fields face. These challenges show STEM disciplines for what they are - intrinsically interesting and deeply relevant to our lives. In other words, reflecting on the ethics of STEM can make computer science, engineering, etc. less dry and technical. Students that would have found such a one-sided view off putting may look at a career in STEM differently because of the social dimension that is invariably emphasized in ethics discussions. Including ethics in STEM education may, quite simply, serve as a recruiting tool. This is a possibility worth pursuing given the current need for more STEM professionals.

Related work

Pokorny [4] describes his workshop that he presented at the CSTA (Computer Science Teacher Association) sessions on Scratch programming, computer hardware, robotics, etc. Participants in his workshop were involved in computational thinking, which added a unique aspect to the workshop. Because enrolment in computer science related programs during the was 60%, which is relatively low compared to the previous years, the CS4HS program was introduced in the university to help students learn about the field of computer science and to increase computational thinking. The goal of this program was to enlighten the teachers beyond computer programming, thereby increasing each student’s interest in computer science. Reaching out to the pre-college audience with regards to computational thinking was crucial. The workshop focused on the idea that computational thinking (CT) is a fundamental skill for everyone and not restricted to computer scientists.

There were several unique aspects incorporated in the CPATH workshop. First, elementary, middle school and high-school teachers of any discipline were invited to participate in the presentation and activities related to computer science and computational thinking. Because computational thinking is relatively unheard of in middle schools and high schools, and because high school Computer Science programs are on a decline, a significant time was spent to help the participants create an action plan to incorporate computational thinking in their respective school. The K-12 educators need to be exposed to the excitement and power of computer science and computational thinking. There were goals established in order to achieve this. The first goal was for the teachers, administrators, and students to understand the discipline computer science. The second goal was to promote K-12 computing curriculum. The third goal was to encourage the teachers to incorporate the computational thinking concepts into their curricula.

The presentation covered the required technical skills and led particpants through hands-on activities to help the participants better understand computer science and computational thinking. Hands-on activities helped the participants to enrich their knowledge on computer programming, algorithm development and problem solving strategies, beyond standard technology courses. A goal of the workshop was to recruit students into computing majors. The primary funding source was from Google, which included completing an online proposal, submitting an estimated budget, and describing the participants/audience. In addition to a small stipend, the participants earned a teaching certificate from CPDU (Continuing Professional Development Unit).The participants were surveyed to indicate the overall satisfaction of the workshop through open-ended questions. The format of the workshop was generally liked by the participants with respect to hands-on activities, guest speakers, and free resources and materials directly applicable to the respective class.

The workshop was a great growth experience in terms of the participants gaining knowledge and understanding about the concepts of computer education. The workshop created a great impact on the pre-college audience by presenting a good view of computing career opportunities and hopefully increased the interests in the fields of computer science among all the participating students.

Barr and Stephenson [5] suggest that computational thinking enables researchers to advance in computing with regards to new problem solving strategies and testing new solution in both real and virtual applications. The students would be required to work with problem solving algorithms, computation methods, and tools in K-12. The K-12 curriculum, they say, must focus the concepts and efforts in two directions. Initially a change in education policy must be made, and, later, K-12 teachers need resources and appropriate examples to teach the students. The gap between K-12 and computer science education communities has to be bridged. The goal is to absorb the key concepts of computation to be applied across varies discipline rather than focusing on a specific computer science field. The computer science education community can help to integrate application of computation methods and tools across diversified fields of learning. This could be accomplished in parallel with the computer science understanding of the K-12 education setting and implementing that knowledge in to research activities.

Computer scientists can provide an understanding of how the computational process affects other fields and also how common problems affect various disciplines. Computer science is a field that consists of mechanics, design principles, and practices. The Association of Computing Machinery (ACM) model curriculum for K-12 computer science defines computer science in relation to programming, algorithmic process, hardware and software design, and its impact on society. Computational thinking has numerous elements in common with other fields like algorithmic thinking, engineering thinking, and mathematical thinking; however, unique to computational thinking are the construction models, finding and correcting errors, creating representations, and analyzing concepts. The ultimate goal is to apply common elements to solve problems and invent new equations rather than teach everyone to think like a computer scientist.

K-12 education is currently highly complicated, involving multiple computing priorities and ideologies. In order to achieve systemic change in this environment, a thorough understanding of the entire system is a must. However to incorporate computational thinking into a practical K-12 approach, an operational definition is also required. Activities like modeling, simulation abstraction, and automation involve iterative design and refinement. CSTA (Computer Science Teachers Association) and ISTE (International Society for Technology in Education) aimed at promoting an operational definition of computational thinking for K-12. CT is a method used for solving problems as tool builders; it is also an approach that can be automated, transferred, and applied across diversified fields.

Possible strategies to achieve systematic change would be policies, vision, and language related to the K-12 educational policy, and inspiration and leadership qualities. K-12 educators play an important role in improving student learning because of their expertise and their knowledge of resources at all levels of education.

Amber Settle, et. al. [6] describe computational thinking as an emerging field comprised of basic skills that should become a core part of education. They say that computational thinking is one of the most important areas in determining one's analytical abilities. Using the process of iterative design, refinement, and reflection, it is possible to integrate computational thinking into six courses: middle-school computer science, high-school computer science, high-school Latin, graphic arts, English, and history. Computational thinking involves a wide set of techniques and skills adopting 'great principles of computing,' which includes computation, communication coordination, recollection, automation, evaluation, and design. Based on the pedagogical approach, the programming skills of students can be measured using an on-line system. A student's performance, ability, confidence, or stress can be noted. Ultimately, a student would be tested for computational thinking based on abstract thinking, application of programming concepts, and computing solutions to problems.

Sheikh Iqbal, et. al. [7] states that the wide exposures of computational thinking in various fields of natural sciences have attracted many new students into the computer science domain. In recent years, the increased need for trained computer scientists with computational thinking skills is tremendous. The workshop conducted on computational thinking has its influence mainly related to the deep connection between natural science, mathematics, and computer science. The Python programming language was emphasized to demonstrate simulations and computations. The modules for the workshop were designed for common areas like probability, simulation, and computational thinking. The technical sessions concentrated on the concepts in the scientific domain through a combination of hands-on activities and simulation development. The sessions on computational thinking particularly involved the concepts of abstraction, algorithm, simulation, and the sciences. The participants were also given assignments on developing algorithms for day-to-day common activities like making peanut butter and jelly sandwiches. They were also introduced to a wide verity of simulation concepts and techniques to differentiate deterministic simulations.

If computational thinking concepts were introduced to high school teachers, a remarkable change could take place with respect to a long-term perception about computer science and other fields related to it. Workshops in these areas would enhance the materials used in high school curriculum. Teacher and student training on computer programming would enable them to be effective in performing logics, functions, graphics, and analysis for any real world application.

Amber Settle [8] describes the workshop conducted by an NSF-funded project focusing on computational thinking as an important part of the curriculum. With recent developments in the field of information technology, educators have identified computational thinking as vital to computing curriculum. In order to formulate problems and their solutions, wide knowledge in computational thinking is essential; teacher workshops help to increase this knowledge about computational thinking.

Vicki Allan, et. al. [9] states that the rate of enrolment in undergraduate computer science programs has declined during recent years. Efforts were made to make curriculum revisions and redesign undergraduate computer science AP courses at many institutions. Computational thinking is one of the main areas to be included in redesigns. Such inclusion raises awareness of computer science even at the high school level. Some common questions are what materials to include, how to resolve software installation issues, how to deal with different problem-solving interests among students, and how to keep the students interested in creativity and visualization. Thus all the viewpoints and support merge together to promote high state standards and goals on computational thinking in local schools.

Approach followed

In the CPATH project the vertical integration of K-12 students in computational thinking activities was the main objective. This was achieved by splitting up the environmental control project into components for each of the class levels (middle-school, high-school, and undergraduate). The detailed description of the environmental control project is described below.

Description of the Environmental Control System

Students were told that the environmental control system small-screen application that they would be designing would be presented to the engineers who directed the Students Shaping America's Next Spacecraft (SSANS) Program at Johnson Space Center (JSC) NASA. The university STEM staff had previously worked with the SSANS engineers, engaging high school and undergraduate students in designing mock interior components for the Orion Mock-up Space Capsule. The SSANS and the Orion Mock-up Space Capsule was the inspiration for the environmental control system for small-screen application, and the STEM staff presented of the application to the SSANS NASA engineers in the spring 2012 semester.

The student participating in the project were told that the environmental control system for the Orion Mock-up Capsule would be used to control and monitor environmental variables such as temperature, oxygen, carbon dioxide, and humidity. Students were guided to ask questions about what kinds of effects changes in temperature, humidity, oxygen, etc. might have on astronauts in the capsule. The students were encouraged to consider not only the physical effects but psychological effects as well; the latter can be critical on prolonged space missions. The environmental control system would need to be equipped with sensors to measure the values of the variables. See Figure 1 below.

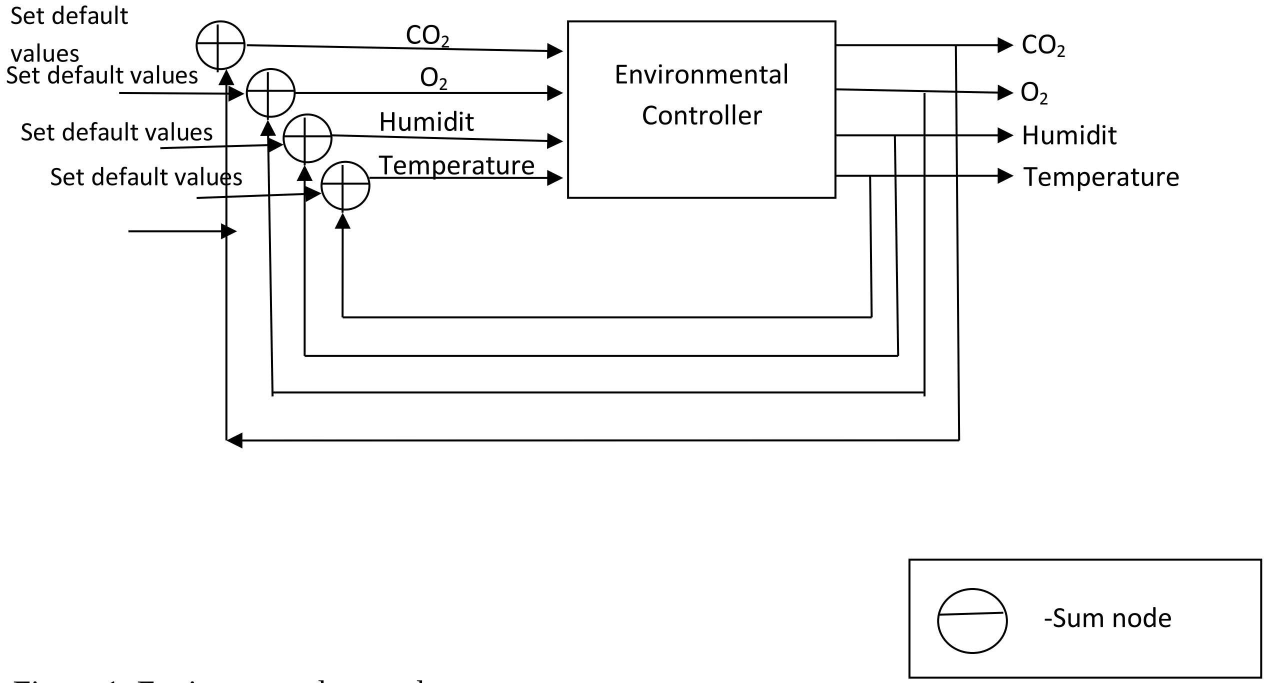

Figure 1. Environmental control system

The environmental control system is a closed loop control system; i.e., it uses feedback to regulate changes to the transfer module's environmental control system. In Figure 1 the "sum" node is used to compute the difference between the values the system actually measures in Orion Mock-up Capsule, the reference values NASA SSANS engineers set, and the environmental variables in the Orion Mock-up Capsule module; this difference is given as input to the controller. The controller then sends appropriate control signals to minimize the difference and bring the environmental variables to the desired reference values. For instance, if the temperature set-point value is 70°F, and the sensors report the temperature value as 80°F, the difference value computed by the difference node results in the controller turning on the air conditioner to lower the temperature until the measured value is equal to the reference value; thus, the controller is used to regulate the environmental variables by either decreasing or increasing their values to match the desired reference values.

Additional information

- If the value of any environmental variable falls well below a specified minimum threshold or increases beyond a maximum threshold, an alarm can be activated to notify the humans on board in the mock-up space capsule; this feature will alert the humans in the space capsule that there might be an emergency.
- Graphical representation of the history of values for the different environmental variables obtained using bar charts, line charts, etc., may be desirable.

The project components entitled to students were in the form of computational thinking activities to make it more understandable and implementable by the students. The middle school students were given four activities such as understanding the problem statement, requirements analysis, binary number problem, and weather data collection activity. The high school students were given four activities such as understanding the problem, designing the user interface, technical analysis, and programming using scratch/Alice software. A sample activity which was used in this project to make the student understand the underlying project to be analyzed can be found in Appendix A.

Research questions

While working on this project, the team initially had some research questions that were resolved at the end of the project:

- How can we involve today's technology-driven K-12 students in computational thinking activities?
- Will iPad/iPhone applications developed by undergraduate students be a good tool for middle school and high school students to develop their computational thinking power?
- What kind of involvement will iPad/iPhone applications create among K-12 students and college students?
- How can we create an iPad/iPhone application?
- What can we learn by introducing iPad/iPhone applications inside the classroom?

The above research questions have been addressed through the implementation of the project. The project described in this paper demonstrated how computational thinking activities could be incorporated in the classrooms. The use of iPad/iPhone application development examples among middle school, high school, and undergraduate students was effective in terms of increasing the students' interests in participation.

How the project was implemented

At the university, vertical integration among undergraduate students was achieved by Senior Project Design students sub-tasking tasks like requirements review, design review, and code review to first year (freshmen)students from Programming Principle I class, second year students from Data Structuring class, and forth year students from Software Engineering class. Inspire – CT activities carried out during the Spring semester 2010 were presented at the Inspire-CT faculty workshop held between dates June 7-9, 2010.

The main task to be accomplished for the final and following year of the CPATH project was to vertically integrate and involve middle school students, high school students, and undergraduate university students in computational thinking projects.

The CPATH team chose an environmental control module from the space shuttle control module project for simulation and implementation. For any software development process, major activities that need to be carried out include requirements elicitation/specification, design, implementation, and testing. In this project, middle school students were treated as end-users, who helped in identifying/eliciting requirements for the chosen module of NASA space shuttle project. Mock-ups of the modules developed were created by the middle school students. High school students, who have beginner's knowledge of computing technologies, took into account the requirements collected from the middle school students and were involved in designing and modeling the modules. Senior level undergraduate students, who have mastered the latest technologies, were involved in building an iPhone application that would simulate and implement the modules of NASA space shuttle designed by the high school students.

The overall picture of the system with transition is described in Figure 1.

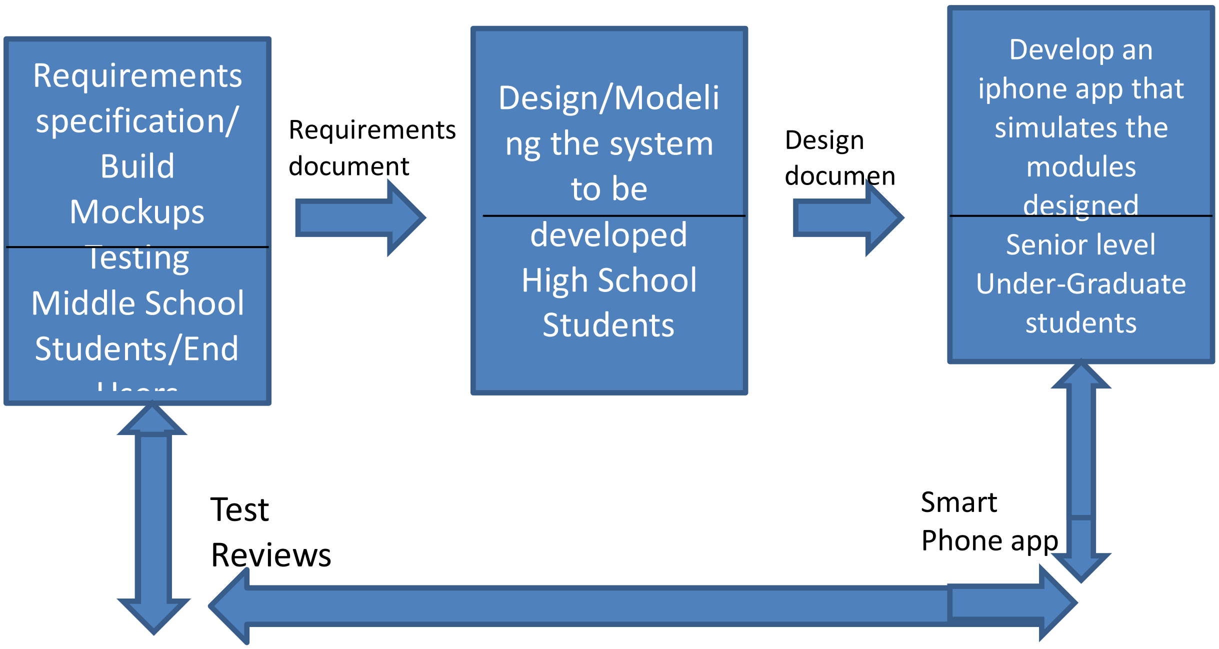

Figure1. Flow of activities among middle school, high school, and undergraduate students

Methodology

Readouts, detailed description, and presentation were distributed to the teachers to educate them on the objective of the project and the list of activities that needed to be accomplished. The module, chosen from the space shuttle project was explained in detail along with the project documentation. Tools and software appropriate for middle school and high school students to elicit requirements and to design/model the NASA modules were identified. Handouts and forms were used to document the progress and to be filled out by the actors (participants) of the system. Once, the requirements and design document has been generated and obtained from the middle school and high school students, the same were passed on to the senior level undergraduate students for implementing the mobile app that simulates the system modules.

Resources involved in the project

The main resources involved in this project are as follows:

- Human resources – Middle school students, high school students, school teachers, school administrative staff, college level graduate students, college level undergraduate students, college professors, college administrative staff, and the university's STEM Center staff.
- Equipments used – computers with necessary software installed, weather station

units, video conferencing tools to communicate among participants, and stationary items such as paper and pencil.

- Documents – The computational thinking activities were developed and were given to school teachers and students involved in the project prior for their easy understanding.

Discussion on the outcomes from the project

The teachers who were involved in the project felt positive about computational thinking concepts and the integration inside their classrooms. In order to conduct computational thinking activities inside a classroom, the teachers and the students need to understand the computational vocabulary early in the project lifecycle. The use of scratch software to teach students programming was well received and enjoyed by the students from different grades. The students enjoyed using the software program and were also able to think about the different variations that go behind the application. This project got the students thinking on a higher level than they were used too. The activities that were developed in a way for the students to be interacting among their peers as well as with undergraduate students got them out of their comfort zone. The project involved a simulation application that controlled environmental parameters. The students were much more involved in developing such an application that had processing information that would affect other people. The students were also concerned for the long term effect of decisions made. The teachers were impressed by the connections made between science, math, and technology through this project. This project gave an opportunity for students to experience the real connection of data to technology. The teachers were able to correlate the computational thinking activities that were carried out along with their regular class curriculum after carrying out the activities.

Challenges faced while executing the project

Some of the challenging factors encountered while carrying out this project are as follows:

- To enable effective communication among the middle school, high school and the college level students since this is a vertically integrated project.
- To effectively convey the concepts and components to the school teachers before taking the concepts down to the students
- To obtain updates and progress status reports from the school students to ensure the progress is in the right track
- To maintain journal in the progress of the project with inputs from all the human participants of the project
- To install the technology needed at schools and at the workplace of the human participants of the project
- This project had the feedback component were outputs from each level are taken back and iterated to obtain efficient results. Such iterations always comes with a delay factor and a time component which is a challenging part
- To effectively tie the components of computational thinking with the course curriculum of the students in accordance with the STEM discipline standards
- To address the ethics component of the project to school students and teachers and also

to get them involved to thick about ethics

The above challenges were resolved as the project progressed. Few of the essential factors that were needed to overcome throughout the project were effective communication, team work, pre-planning, critical thinking, monetary support, and involvement of the participants.

Summary

There were 12 school students and 4 school teachers involved in the CPATH project. Over the project, the K-12 school students and the teachers gained valuable knowledge about computational thinking concepts. The students were well responsive and showed gradual progress, and their interest level and involvement towards the project increased over time. Towards the end of the project, the K-12 students' questions were knowledgeable and thought provoking. Their questions were directed, primarily, to the undergraduate students participating in the project. This was excellent experience for the undergraduate students to have to field questions about computer science and computational thinking from people other than others in the computer science discipline. Teachers, who played an integral part in this project, maintained daily logs on the progress of the project and they integrated computational thinking into some of their other classes, such as mathematics and English. The teachers also documented how the students were engaged in the activities for the projects, the pace at which the students were able to complete the tasks given, and also how well the students adapted to the requirements of the project.

Students who showed minimum interest during the start of the project got involved and interested towards the end of the project and they gained considerable knowledge and experience about computational thinking and computer science. Teachers commented that the students seem to have acquired some higher-level thinking skills and were able to comprehend complex concepts better than before they engaged in the CPATH project. The computational thinking tool introduced in this project helped students to approach problems from different perspectives and to come up with novel and creative solutions. Overall the success of this project was tremendous, and it gave the pilot school a project-based learning project to integrate into its STEM curricula.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. CNS-0939059, 0939088, and 0939028. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Reference

1. Jeannette M. Wing. Computational Thinking. Communications of the ACM, v.49 n.3, March 2006
2. National Society of Professional Engineers Code of Ethics, NSPE, accessed January 4, 2013, http://www.nspe.org/Ethics/CodeofEthics/index.html
3. Association of Computing Machinery Code of Ethics: http://www.acm.org/about/code-of-ethics?searchterm=code+of+ethics

4. Pokorny, Kian L., and Nathan White. "Computational thinking outreach: reaching across the K-12 curriculum." *Journal of Computing Sciences in Colleges* 27.5 (2012): 234-242.
5. Barr, Valerie, and Chris Stephenson. "Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?." *ACM Inroads* 2.1 (2011): 48-54.
6. Settle, Amber, et al. "Infusing computational thinking into the middle-and high-school curriculum." *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education*. ACM, 2012.
7. Ahamed, Sheikh Iqbal, et al. "Computational thinking for the sciences: a three day workshop for high school science teachers." *Proceedings of the 41st ACM technical symposium on Computer science education*. ACM, 2010.
8. Settle, Amber. "The computational thinking across the curriculum workshop." *Proceedings of the 2011 conference on Information technology education*. ACM, 2011.
9. Allan, Vicki, et al. "Computational thinking in high school courses." *Proceedings of the 41st ACM technical symposium on Computer science education*. ACM, 2010.

Appendix A

Activity 1 – Understanding the system

Teacher Section

Computational Thinking Project—NASA Orion Crew Transfer Module

Understanding the Environmental Control System In this computational thinking project, students have the chance to explore computational thinking by designing an automated environmental control system for the Earth to Mars Orion Crew Transfer Module. You can find information about the module at www.nasa.gov/exploration/systems/mpcv/index.html.

NASA built the Orion multivpurpose crew vehicle (MPCV) based on the design of the constellation program. The Orion Crew and Service module (CSM) consists of two parts: a crew module (CM) and a cylindrical service module (SM). The Orion CM is designed to hold four to six astronauts and supports an auto dock feature that allows the flight crew to take over in case of emergency. It is the transportation capsule that will provide a habitat for the crew and storage for consumables and research instruments, and it serves as a docking port for crew transfer. The CM returns to earth after each mission, and this is the only part of the

MPCV that returns to earth. The CM has been designed for 21-day flights, which could be expanded in future.

In this NASA Orion Mock-up Capsule project either divide the class into end users and designers, and use this activity for the end users; or treat the whole class as end users. You might also consider using different classes. When we designed the course, we asked middle school students to be end users, high school students to be designers, and college undergraduate students to be developers.

Activity 1

Understanding the Environmental Control System

Summary

This activity focuses on helping the students understand the system to be developed.

Skills required

Understanding and following the instructions delivered.

Ages

Middle school and high school students

Materials required

- Project description document
- Live demonstration of the system to be developed
- Understanding the System—Feedback Form
- Pencil and eraser

Instructions

- Explain the environmental control system to the students. Review the questions students will receive to be sure you cover everything.
- Play the live demonstration of the environmental control system. [Link to the excel worksheet]
- Let the students ask questions and brainstorm their ideas.
- Hand out the Understanding the System—Feedback Form and ask the students to fill it out. Collect the completed forms for evaluation.

Student Section

Activity 1

Understanding the System—Feedback Form

1. Define environmental variables.

2. Out of the following options choose the environmental variables that need to be controlled in the space station.

a. Temperature

b. Humidity

c. Carbon

d. Carbon-dioxide

e. Oxygen

3. How does a closed loop feedback controller work?

4. What is the list of actions that needs to be performed by the controller?

5. What kind of alarms can be used with the environmental controller to alert humans? What kinds of factors or considerations should be taken into account when choosing an alarm? Do different alarms have different benefits and trade-offs?

Appendix B

Group Activity

Scenario:

One month into the approximately 6 month trip to Mars, a malfunction in the environmental control system for the Orion capsule has led to the temperature in the capsule being stuck at 87 degrees Fahrenheit. The temperature in the capsule does not pose any serious physical health risks, but has made things fairly uncomfortable.

The environmental control system has a back-up system that can be activated to normalize the temperature in the capsule. However, doing so will require that power be drawn from a storage unit containing samples for an important agricultural experiment that was to be conducted on the mission. The samples need to be constantly cooled until they are tested on Mars. The experiment, if successful, would not only provide important data for agricultural scientists on Earth but would also establish an agricultural base (e.g. a diversity of foods to eat) for current Mars crew members as well as future crews. There are enough food supplies (meals-ready-to-eat) for the entire mission even if the experiment is not conducted.

Characters:

- Astronaut - Scientist (on Earth)
- Communicator - Mission Control (decision maker)

Instructions:

1. Students will be broken into 4 roughly equal groups. Each group will have at least 2 high school and 2 middle school students.
2. Each group member will be assigned one of four roles. Each group member will be asked to think about how this scenario affects their role in particular. Emphasis should be placed on developing an exhaustive list of considerations as well as an initial decision about what to do (e.g. draw power or don’t draw power). Students will be asked to generate this list quietly and on their own (5 minutes)
3. The roles will then be separated (in two rooms if possible). Astronauts will be placed far enough away from Mission Control and the Scientist so as to not be able to hear any conversation.
4. The Communicator will be asked to act as an intermediary between:
a. Mission Control and the Scientist
b. Mission Control and the Astronaut
5. The Communicator cannot tell the Astronaut what the Scientist said or vice versa. The time it takes the Communicator to listen to each role and convey that information is intended to mimic potential message delays as the Orion craft gets further and further away from Earth.

6. After 15 minutes, Mission Control will be asked to make a decision. They will be expected to communicate, via the Communicator, this decision to both the Scientist and the Astronaut.
7. DISCUSSION: class will reconvene to discuss the decisions that each Mission Control member made. Each MC will be asked to state his/her decision as well as explain why he/she made that decision.
8. WRITE-UP: each team member will be asked to complete a write up addressing the following (to be completed outside of class):

a. Identify:

i. Name

ii. Grade

iii. Role assigned

b. Respond to the following questions:

i. What challenges did you face in your role?

ii. Did these challenges have an ethical dimension? Please explain your answer.

iii. What were some of the challenges faced by people in other roles in the scenario?

iv. What was the ethically correct decision to make in this scenario regarding the temperature in the capsule? Please explain your answer.

v. Did you change your mind about what you thought was the right thing to do as the activity progressed? If so, why? If not, why not?