STUDIOS & SUSTAINABILITY: A CREATIVE CDIO APPROACH TO COMPUTER ENGINEERING EDUCATION

Emily Marasco, Mohammad Moshirpour, Laleh Behjat, William Rosehart

Electrical and Computer Engineering, Schulich School of Engineering, University of Calgary

ABSTRACT

As creativity and innovation become increasingly important in solving future electrical and computer engineering challenges, such as the end of Moore's Law, engineering education will need to incorporate creative and global problem-solving with technical course concepts. Creative skills and problem-solving lead to new and useful engineering processes, tools, technical and products, and development of these skills should begin early within an engineering program. Creativity within engineering encompasses a wide variety of definitions and aspects, including integration on a global scale and sustainability engineering development. Sustainability is an element of creative design that is often emphasized in other engineering disciplines while being overlooked in electrical and computer engineering. However, past research has shown that students do not view engineering as a creative career or one that allows them to make a sustainable impact on society. This paper will detail and explore the impact of a flipped classroom pilot for a required first year engineering computer programming course, and effects on its implementation on student perceptions of engineering. This initiative uses the CDIO method to integrate sustainability, design, creativity, arts and entrepreneurship through hands-on studio experience.

This course initiative requires students to build a creative and visual-based programming portfolio using the Processing programming language, working towards the development of a final coding product. The evolution of this final product requires students to apply the four CDIO stages to either a bio-inspired visualization project or a game creation project, while incorporating elements of sustainable engineering design. A final demonstration and peer assessment also introduces an element of entrepreneurship and business presentation.

This paper will describe the studio learning experience, the integration of the CDIO method with technical coding objectives, and preliminary results and observations from the fall semester offering. Overall, students have demonstrated more engagement and motivation due to the creative CDIO design techniques introduced in parallel with technical programming concepts. The developed curriculum may also be modified for a K-12 introduction to computer engineering and programming as computer literacy becomes more prevalent within schools.

KEYWORDS

Creativity, Sustainability, Programming, Flipped Classroom, Standards: 2, 3, 5, 6, 8, 11

COURSE OVERVIEW

Creativity is an important part of engineering graduate attributes across the world, including the attributes required by the Canadian Engineering Accreditation Board (CEAB, 2014, IEA, 2013). While there are varied theories and models of creativity (Taylor, 1975, Guilford, 1967, Torrance, 1974), this work uses the Taylor hierarchy of creativity (Taylor, 1975) to define creativity as consecutive stages of understanding and the development of creative thinking skills. In many engineering programs, students are expected to develop their innovation and creativity through open-ended introductory and senior design courses, while technical courses remain focus on domain knowledge. Students themselves are often focused on fulfilling assessment requirements necessary to achieve a high grade, sacrificing creative knowledge application in favour of a perceived correct answer. Despite an emphasis on creativity in industry hiring, related literature questions whether or not creative thinking is truly being taught in engineering postsecondary programs (Daly, 2014). Existing literature identifies a clear need for creativity to become a greater focus in engineering education so that graduating engineers are capable of meeting the innovation challenges of the future (Daly, 2014, Felder, 1987, Felder, 1988, Liu, 2004, Joseph). Previous CDIO initiatives (Arboelda, Jingdong, 2011, Shen, 2009, Chunfang, 2012) demonstrate work in the use of the CDIO framework for computer engineering education, while this research builds on the previous literature by combining the development of both creative and technical learning outcomes within a traditionally technical course, rather than a design course.

This work describes and details an initiative to redesign the mandatory first year engineering introductory programming course, required by nearly 800 students in a single semester, using the CDIO framework and syllabus attributes. This course was redesigned as a flipped classroom, where the instructional content was provided online through various mediums, while the instructor directed portion was delivered through exploratory, hands-on exercises. This paper will focus on the exploratory design portion of the course.

STUDIO EXPERIENCE

Like learning a second language, computer programming education requires not just the understanding of vocabulary and syntax structure, but the application of those concepts within problem-solving contexts as needed. In arts education, a studio environment is commonly used to allow students to collaborate with their peers in an open space, while still working individually on their creative product. Within studio sessions, students are often provided with creative prompts, giving them a starting place for creative design. The form of this prompt may vary depending on the medium. For example, a few written lines of dialogue may be given to start a creative writing piece, or a still-live model to start a sketch. Creative spaces are now being explored in engineering institutions as a way to the encourage teamwork and problem-solving required by national engineering graduate attributes (Zabudsky, 2015).

This initiative adapted arts education delivery approaches for encouraging peer cooperation and a deeper, hands-on understanding of technical concepts, allowing for development within CDIO attributes such as creative thinking, critical thinking, analysis, teamwork, and communication, alongside development of computer science and programming knowledge Engineering studio environments are proposed as an alternative to laboratory settings for collaborative peer technical practice, alongside project or assignment prompts that encourage students to start the creative thinking process.

Studio Workspace

Without the need for a traditional lecture hall, this scheduled portion of this course was run in the open workshop spaces usually reserved for the first year design course. Students were seated around tables in groups of 4, with an average of 32 students in each of the six available workshop rooms. To accommodate nearly 800 students, four separate studio session times were run. Each studio room was supervised and directed by a teaching assistant, under the supervision of a teaching assistant coordinator and the two course instructors. Students met in this studio space once a week for two hours to complete exploratory assignments and to work on their final design project.

Studio Activities

The studio activities and assignments allowed the students to reinforce their online learning while exploring the application of theoretical concepts. The assignments were designed to build in complexity and difficulty each week, resulting in a final coding portfolio that could be demonstrated to a future employer or internship opportunity. Each portfolio assignment consisted of an in-studio and post-studio component. The in-studio exercises were designed to be shorter and collaborative, allowing students to work together on a combination of written and coding problems. These exercises were also used as an attendance tool, as they were required to be submitted in-person before the end of the studio session. The post-studio exercises were due one week later, submitted online for further evaluation.

SUSTAINABILITY IN COMPUTER ENGINEERING

Sustainability is an important part of engineering design education, included in CDIO Syllabus 4.1.7 Sustainability and the Need for Sustainable Development, but is often not addressed in electrical and computer engineering projects. As this course was an introductory course for computer and software engineer concepts, sustainability was included as an integral part of the introductory education.

Sustainability Compass

The University of Calgary promotes the four primary categories of the Sustainability Compass, commonly used to identify different key issues and aspects of sustainability (University of Calgary, 2015). These categories and related concepts were integrated with the course curriculum to demonstrate the impact of technology on sustainability concerns.

Nature

The nature category includes any sustainability issue that may impact the natural systems on which all life depends. Students were educated on related ICT issues, including the environmental impact of e-waste disposal and the impact of technology on natural human and animal behaviours and habits.

Economy

The economy category includes issues relating to the sustainability of economic practices and standards that provide humanity with goods, services, and work. For example, the adoption of automation technology has both created and removed jobs in certain industry.

Software and digital developments, such as bitcoin, may also impact the sustainability of current economic practices.

Society

The society category examines the sustainability of social and cultural systems, from school communities to cultural communities. Students were given many examples regarding the balance between ethics and business practices, such as cybercrime and illegal hacking for purported moral reasons. Another relevant example provided was the use of telecommuting within industry, which requires balance between responsibility, time, effectiveness, and job scope creep.

Wellbeing

The wellbeing category involves aspects that impact individual health, happiness and quality of life. Students can relate most to the software-related examples in this category, which include issues such as online bullying, social media harassment, and catfishing. Students are also familiar with the positive elements of sustainable wellbeing practices, including donation campaigns and social media charity challenges.

CREATIVE DESIGN PROJECT

The final creative design project was developed using the CDIO framework to support the integration of technical knowledge, creative development, entrepreneurship and sustainability, pulling together elements from all four UNESCO pillars of education and the CDIO syllabus. Students worked in pairs to develop their choice of either a game or a bio-inspired data visualization project. Over 95% of the students chose to develop a game. The project requirements included some mandatory technical elements that had to be demonstrated, but otherwise students were given complete creative control over their final product. Students were required to demonstrate evidence of beta testing and documentation throughout the coding process, giving them the opportunity to integrate feedback from their peers and teaching assistants.

Creative Development

The project outline followed the CDIO design process, allowing the students to develop their ideas in an iterative and detailed manner. The design deliverables used brainstorming, visual mapping, creative storyboarding, and synthesis techniques to specifically develop the attributes found in CDIO Syllabus 2.4, especially 2.4.3 Creative Thinking.

Providing students with the chance to visualize their program and ideas allowed them to isolate any potential issues with user interaction, continuity, code layout, etc. Storyboarding is a technique used to map the intended stages of a storyline. This allows the creators to identify changes in character, dialogue, mood, scenery, and style. Each storyboard scene consists of a drawing or sketch, sometimes with a few lines of explanation or dialog written underneath, as shown in Figure 1. A storyboard should be divided into enough scenes to represent the entire process without missing any significant component. This technique allowed students to conceive each design stage and aided project management planning.

**Sketch/diagram of game screenshot goes here.
This diagram should show what is happening on screen during a particular moment or scene in the game (i.e. Level 1 gameplay, or Level 2 score calculation, etc.)**

Caption: One word summary of diagram content.

Explanation: Include a more detailed description of this particular moment or scene in the game, why is is important, and how it works.

Storyboards may include things such as:

- Each available level
- Menus
- Scoring
- Plot points/cinematic scenes

Use your storyboard to outline each interaction the user will have with your game. A multi-level game with a menu and rules screen will probably have more than six scenes in its storyboard.

Figure 1: A storyboard layout allows students to conceive and plan each stage of their design.

Integration of Sustainability

The project was designed to include a final report and documentation that required reflection and clear thought regarding the applicability of ICT sustainability issues. Students reported on various sustainability aspects, from physical ergonomic issues in gameplay to e-waste education, and from responsible program messaging to the accessibility requirements of disabled users.

Demonstration and Assessment

The final project culminated in an exhibition and demonstration day, where students were given the opportunity to test and experience the final products of their peers. This activity included a peer assessment assignment, where students were required to provide feedback and a rating for the other teams in their studio room. In their final reports, students were also required to reflect on their own experience within the project, including team dynamics and project management elements. This was designed to support the development of CDIO attribute 2.4.5 Self-Awareness, Metacognition and Knowledge Integration.

IMPLEMENTATION RESULTS

A preliminary analysis of the course implementation results has been successful, and will be used to further development the next iteration of the course curriculum. Feedback methods included team surveys and reflections, individual surveys before and after the final design project, and university required course ratings. All of this information has just recently become available following the completion of deferred examinations, and a further analysis of the information will be included in the final paper.

Overall, students reported the course to be similar in workload or slightly higher when compared to their other first year courses.

Creative Project Response

Students reflected on the creative design project with both qualitative and quantitative feedback. As shown in Table 1, the majority of students agreed that their understanding of technical concepts was increased through the creative design project, as well as their creative thinking. Over half of the students reported an increase in both their interest and enjoyment of programming as well. The majority of the students also reported learning additional concepts on their own time in order to the complete the project. When asked about this question, most of the students indicated that this was not due to a lack of understanding or lack of appropriate course content, but because their creative designs and desire to produce a particular idea required additional research of concepts outside the scope of the course.

Table 1: A summary of student response towards the creative design project.

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
My understanding of technical concepts increased	2.02%	5.26%	12.15%	52.63%	27.94%
My creative thinking was improved	1.62%	5.26%	18.22%	48.18%	26.72%
My interest in programming increased	9.31%	8.50%	17.41%	35.63%	29.15%
My enjoyment of programming increased	8.61%	8.61%	16.39%	37.70%	28.69%
I learned additional concepts on my own to complete the project	2.83%	2.43%	9.72%	46.96%	38.06%

Student Perceptions

Students were also asked about their perceptions of electrical, computer, and software engineering as a result of their participation in the course. 70% of the students agreed or strongly agreed that programming is an important skill for some engineers, but more importantly, only 25% disagreed that programming in an important skill of all engineers. With the increasing focus of computer literacy and programming knowledge, it is important that students recognize the impact that coding skills and problem-solving can have on engineering innovation and development.

Student Quote

"I believe that the game project allowed students to be creative and innovative. This project was very open ended which I believe was an advantage. Ultimately, I believe that this course was very well executed and organized. This course requires individuals to become better problem solvers, not just plug and chug."

- ENGG 233 F15 Student

CONCLUSIONS

In summary, this work outlined the exploratory design elements of a course redesign initiative for an introductory engineering programming course. The creative design exercises and project were developed using the CDIO framework and syllabus for the effective teaching and learning of creative attributes alongside technical knowledge. Preliminary analysis of both qualitative and quantitative feedback has shown results supporting the desired outcomes of the course redesign, including increased creative thinking and understanding technical concepts. With the co-development of both creativity and technical knowledge, students will be equipped with the skills necessary for future innovation and design.

REFERENCES

Arboleda, H., Pena, E., and Castaneda, L. Engaging engineering students with an introductory CDIO-based robot programming course.

Canadian Engineering Accreditation Board. (2014). Accreditation criteria and procedures. Technical report, Engineers Canada.

Chunfang, H. and Li, L. (2012). Research on CDIO and the influence of school-enterprise co-operation in the CDIO. In The 7th International Conference on Computer Science and Education. Melbourne, Australia, 2012.

Daly, S., Mosyjowski, E. and Seifert, C. (2014). Teaching creativity in engineering courses. Journal of Engineering Education, vol. 103, no. 3, pp. 417-449.

Felder, R. M. (1987). On creating creative engineers. Engineering Education, vol. 77, no. 4, pp. 222-227.

Felder, R. M. (1988). Creativity in engineering education. Chemical Engineering Education.

Guilford, J. P. (1967). The Nature of Human Intelligence. McGraw-Hill, New York.

International Engineering Alliance. (2013). Graduate attributes and professional competencies. Technical report, International Engineering Alliance.

Jingdong, S., Zhengbin, W., and Yan, Z. (2011). Computer engineering education reform based on CDIO. In The 6th International Conference on Computer Science and Education. Singapore.

Joseph, B. Engineering creativity peaks in kindergarten. Website. http://www.designnews.com/.

Liu, Z. and Schonwetter, D. (2004). Teaching creativity in engineering. International Journal of Engineering Education, vol. 20, no. 5, pp. 801-808.

Shen, Y., Xue, H., Zhang, M., and Fan, Y. (2009). Exploration and practice of CDIO educational idea in teaching reform of computer foundation. In Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on, pages 1-3.

Taylor, I A. (1975). Perspectives in Creativity: An emerging view of creative actions. Aldine, Chicago.

Torrance, E. P. (1974). Torrance tests of creative thinking: Norms and technical manual. Scholastic Testing Press.

University of Calgary. (2015). Defining sustainability: Sustainability Compass. University website report. http://www.ucalgary.ca/sustainability/home/defining-sustainability

Zabudsky, J., Rayegani, F. and Katz, Y. (2015). The power of creative space in engineering education. In Proc. of the 11th International CDIO Conference.

BIOGRAPHICAL INFORMATION

Emily Marasco is a Ph.D. candidate at the Schulich School of Engineering, University of Calgary. Her research focuses on creativity and cross-disciplinary curriculum development for engineering students, as well as for K-12 and community outreach programs. She is the co-founder and chair of the university Engineering Education Students' Society.

Mohammad Moshirpour is a software engineering postdoctoral research at the Schulich School of Engineering, University of Calgary, with experience in software development and engineering education. He is an active member of IEEE in southern Alberta and is a regular sessional instructor across various courses and levels of undergraduate engineering.

Laleh Behjat is an Associate Professor at the Schulich School of Engineering, University of Calgary. Her research interests are focused in engineering education, VLSI design, and electronic design automation. She has taught a range of courses and regularly implements active and experiential learning as a tool to improve teaching and learning. She has consistently received high student and course evaluations, and has been nominated for a Student Union Teaching Award.

William Rosehart is the current Dean of Engineering at the Schulich School of Engineering, University of Calgary. His research areas include engineering education, optimizing energy systems, controlling wind-based energy production, and studying the benefits of distributed generation. He has received a number of teaching awards and early career awards, including recognition as one of Calgary's 2007 Top 40 under 40.

Corresponding author

Ms. Emily Marasco
Schulich School of Engineering
University of Calgary
2500 University Drive NW
Calgary, Alberta, Canada
T2N 1N4
1-403-210-8901
eamarasc@ucalgary.ca