Imagining Creative Hands-On Projects for Students in Computing in the Arts and STEM Incubator Programs

Jennifer Burg
Wake Forest University
Winston-Salem, NC
(336) 758-4465
burg@wfu.edu

Abstract

This paper offers a perspective on the present and future of STEM education. The author begins by describing her current work: NSF grant DUE1323593, entitled "Computing in the Arts—A Community Building Initiative," and an entry-level hands-on learning program at Wake Forest University called the STEM Incubator. The NSF grant enables the principal investigators to foster *computing in the arts* (CITA) programs by means of three yearly workshops and shared curriculum material. The author's original curriculum material provides resources for interdisciplinary courses that intertwine art, music, and computer science perspectives. The STEM Incubator relates to the CITA effort in that it provides an environment where students with diverse backgrounds can collaborate on hands-on projects inspired by art or responding to human needs. To place this work in context, the paper reviews trends of the last decade, a time when constructivism, situated learning, and interdisciplinary collaboration were of prominent interest, while maker spaces and tinkering labs were sprouting up all around. To illustrate the type of work that might engage students in STEM Incubator-type courses, three student projects are imagined, and the rewards and challenges of interdisciplinary, hands-on work are discussed. The projects illustrate how students might be attracted to purposeful STEM learning that combines art, music, artistic and computational creativity, science, and mathematics. The reader is invited to consider just how far "want-to-know" learning might take us if can unleash the synergy between art and science, engage students in playful and creative exploration, and tolerate open-endedness and "teaching without a net."

Goal and methods

The NSF TUES grant described in this paper involves three universities: College of Charleston, the University of North Carolina at Asheville, and Wake Forest. Each institution holds a separate grant but the three share an overarching goal: to foster Computing in the Arts (CITA) programs at other colleges and universities and create a community of educators who share ideas and practices. The key activities and method of achieving our goal are three workshops in which artists, musicians, and computer scientists gather to think concretely about what constitutes a successful CITA program, and to generate and share project ideas and curriculum material. The Wake Forest PI's personal activities in the grant involve the creation of courses, projects, and curriculum material that attract student engagement in STEM through computation related to digital images and sound.

For its part, the STEM Incubator aims to attract entry-level students – particularly women – into computer science by means of engaging, socially-relevant, interdisciplinary hands-on projects. STEM Incubator courses were initiated in the fall of 2014. These courses (one-credit each) are offered in a variety of areas: e.g., bioinformatics; digital imaging or music; 3D printing; innovative input/output devices for those with special needs; drones; robotics; and so forth. The courses follows a mentor/apprentice model, with entry level students (apprentices) learning from more advanced students (mentors) in projects that can progress in successive semesters. A student can take the course three times if the topic varies, twice as an apprentice and once as a mentor, or vice versa. The three STEM Incubator credits can count toward a major or BA in computer science degree. In the projects undertaken in the STEM Incubator courses, students are encouraged to dream up something that they'd like to do, with no concern, at the outset, about whether it's feasible or it's been done before. Instructors offer suggestions and sometimes have on-going projects that entry level students can join as apprentices. Projects are often motivated by their usefulness for those in need, or by creative, artistic impulses. Learning emerges from exploration and experimentation in a kind of "want-to-know" learning model.

Outcomes

Key outcomes from the grant have included the workshop results posted at http://compsci.cofc.edu/nsf-cita-workshop.php; publication of talks and other material linked from the workshop site; a SIGCSE conference publication regarding Wake Forest's STEM Incubator; and textbooks written by two of the grant PIs linking computation and music. Approximately 50 art and science educators from widespread universities have attended our workshops thus far. The textbook entitled *Digital Sound and Music*, with interactive learning supplements, is at digitalsoundandmusic.com. Over 2000 visitors a month, from all over the world, view the *Digital Sound and Music* website. The book will be published in hard copy by Franklin Beedle in March 2016. It will remain free to the public online. Future work includes a May 2016 workshop and additional interdisciplinary project ideas to be published online. The key outcome from the STEM Incubator initiative is the number of female students involved in the courses since 2014, which has risen to 50% Table 1 shows the demographics of the STEM Incubator course (where course sections covering different topics are combined).

Table 1 Demographics of STEM Incubator Courses

Semester	Total # students	Male	Female
Fall 2013	30	18 60%	12 40%
Spr 2014	46	27 59%	19 41%
Fall 2014	47	30 64%	17 36%
Spr 2015	23	12 52%	11 48%
Fall 2015	30	15 50%	15 50%

Broader impact

The interdisciplinary nature of CITA programs inherently gives them broad impact. Computing in the arts stimulates STEM learning by means of creative expression, an approach has been found to broaden opportunities in computing for women and minorities. The approach interweaves the computer and arts populations and their bodies of knowledge, bringing more balance in terms of gender and ethnicity in computing. The infusion of the artist's perspective into creative computation enriches and diversifies STEM education, attracting a wider variety of mind-styles into synergistic, purposeful learning. The broadest impact of the NSF-supported work lies in the influence that educators can have on today's students as they learn to live rich, fulfilled, intellectually challenged, caring lives in a fast-changing, closely networked, digitally-dominated world. In hands-on, experimental courses or projects, students can be set free to follow their imaginations, but with the attentive and pre-meditated guidance of their professor, whose role is to nudge the students in fruitful directions. In this way, students can experience a broader perspective on how others think, can deepen their understanding of mathematics and computer science, and can be encouraged toward more life-enriching uses of their digital devices.

From Then to Now in Computer Science and STEM Education

Before venturing a vision of STEM education's future, it seems appropriate to take a bird's-eye view of where we are, nationally, in computer science education. A review of the literature reveals that we've gone through a decade or more of education research interest in active learning, constructivism, and relevance, with much attention paid to interdisciplinary efforts (Table 2).This emphasis on hands-on learning should come as no surprise in a world where students have their hands on digital devices nearly every waking hour. STEM incubators, maker spaces, and tinkering labs are popping up all over. At the same time, "need-to-know" or "just-in-time learning" seems a natural approach for students who could instantly answer their own questions by "googling it."

Table 2 Common Keywords in SIGCSE, ITICSE, and TOCE papers from 2005 to 2015

	SIGCSE	ITiCSE	TOCE
active learning	100	32	4
situated learning	4	0	1
computational thinking	79	31	4
problem-based learning	9	11	2
interdisciplinary	36	5	1
social or cultural relevance	5	0	0

Abbreviations:
SIGCSE: Conference Proceedings of Special Interest Group on Computer Science Education
ITiCSE: Conference Proceedings of Innovation and Technology in Computer Science Education
TOCE: Transactions on Computing Education

In the computer science literature and NSF grant proposals, *computational thinking* received a great deal of attention, as, educational researchers set about trying to show that a computational approach can provide a new perspective and useful methodology for problem-solving in diverse disciplines. In this climate, interdisciplinary programs and relevance seemed often to go hand-in-hand. Computer science's usefulness in the humanities – literature, dance, music, and art – continued to be explored. Educators came to see computer science's pertinence in almost any other discipline, and applications gained ground over theory.

So here we are, in the midst of a head-spinning digital revolution that has popularized things that were only imagined a few short years ago – voice recognizers that actually understand us and answer our far-flung questions, truly immersive virtual reality goggles, glasses that tell us what we might need to know before we even ask, ubiquitous sensors that anticipate our needs, wristbands that monitor our every move, smart phones, smart prostheses, constant worldwide connectivity, and an app for everything. It seems that computer science education has had to scramble to keep up. While we, with admirable deliberation, teach our data structures and theory of computation courses, the world races forward with transformative, game-changing, even disruptive innovations. Clearly, students who grew up in the digital world will be uninspired by computer science programs that don't give them a tangible, visible connection with today's digital innovations, and a sense that they can contribute something of their own. Thus we have embraced relevance, applications, interdisciplinary collaboration, and hands-on learning.

How well are we doing with this kind of teaching so far? The answer to this question is really just an opinion, a "report from the field." In this spirit, I think it's not inappropriate to shift to the first person for the remainder of this paper.

Imagining STEM Incubator and interdisciplinary projects
I recently spent a week in south central Oregon, and it happened to be the week when I needed to write this paper for the "Envisioning the Future of STEM Education" NSF workshop.

As usual, I spent a good deal of time walking around and taking pictures. I justify this pleasure as "relevant to my work" – the study of images and sound from a digital, computational, mathematical, and (with help) an artistic perspective. Oregon has every green in the 24-bit RGB palette, and then some. The sparseness and lack of clutter in the winter greenery, along with the cloudy sky that surprisingly seems to intensify the colors, makes the Oregon in the winter quite photogenic. Natural fractals greet the camera on every turn – in the fir branches weighted down with the morning's rain, in the fractal tree branches trimmed lushly with fractal-shaped lichen, in the moss dotting the cracked asphalt walkway. I wish that I could document the beauty with pictures in this paper. In any case, these walks have inspired an idea for a student project in one of our STEM Incubator courses. Imagine that I've introduced my students to fractals – self-replicating shapes in which the overall object is composed of smaller versions of the same shape, down to the lowest possible level of resolution. Then here's the challenge to the students;

> Take your cameras and, as a group, go for a long, leisurely walk in a natural setting close to forest, meadows, and streams. Look for fractal structures in nature. You can find them in deciduous tree branches, evergreens, moss, ferns, and so forth. When you find a fractal-like structure, point it out to your fellow students, and discuss what makes it a fractal. Look at the colors and shapes closely, and enjoying nature's art. Then take a picture of it. Take the picture back to the classroom and use it to inspire your own fractal-based artwork. With

the help of your instructor, explore the use of traditional art tools as well as digital photo editors and computer programs operating directly on the pixels.

This project is simple in its way, and no doubt similar projects have been done elsewhere. But it's not a bad idea for a project because fractals have both a computational and visual appeal (and it would work in North Carolina, where I live, which is beautiful as well in winter). I imagine this project for a STEM Incubator course, but it might also work well for a second semester interdisciplinary computer science/art course. If I had an art instructor co-teaching, I'm not sure what elements she would bring in, but I'm interested in finding out. I don't think I could anticipate what a computer science student coupled with an art student might do with this open-ended assignment, but as they explore, I expect I'd have opportunities to explain digital images, pixel dimensions, color representation, and color palettes. I'd also have a rich opportunity to explain recursion to the students in terms of fractals.

A twist on the fractal-based project would be to inject a sound component. The challenge would be to continuously recompute a Julia fractal in real-time, constantly zooming in, but changing the color map of the fractal to reflect the changing frequencies in a piece of music that is playing simultaneously. This project emphasizes the computational component, and it would require the help of a mentor student with some programming experience. It has the advantage of combining visual and computational complexity and lends itself to empirical and theoretical analysis. The first project, however, has the appeal of taking the students back to the ultimate primary source – nature and physical reality – a source that they too often neglect as they stay glued to their digital devices.

Here's another project idea, which came to me last Halloween. Please don't laugh.

> Imagine a dynamically-changing Halloween mask. The mask is shaped around a human face, with a translucent display surface on the front, a light- or heat-sensing surface on the back, and a computational interface to change the display as the wearer desires. The sensors detect the shape of the wearer's eyes, nose, mouth, and eyebrows and project these on the front of the mask. The wearer controls the morphing of these features by a button that indicates a "mood" – happy, sad, angry, surprised, threatening, and so forth. The computer interface morphs the wearer's features accordingly, like the Photo Booth applications that you may have seen. Is this feasible? What would it take to make this mask?

I like this project idea because it has the "fun" element. Admittedly, it's a big challenge and clearly not a realistic goal for an entry-level course, but this makes the project intriguingly open-ended. How would students tackle this challenge, what hope would they have of creating anything significant, and what might they learn in the trying? Flexible display surfaces, small cameras, heat sensors, Bluetooth, small-scale hardware like Arduinos, programming. It's likely that, along the way of their investigations, the students would come up with another, maybe smaller, more feasible idea of their own. It's the learning journey, not the outcome, that's most important.

Challenges
So what would you do if your students wanted to work on one or more of these projects? How would you shepherd them through the projects effectively? This makes for challenging teaching, don't you think? It entails helping students to identify a general goal; determine what they need to know along with the equipment and resources required; determine the extent to which this problem has already been solved by others; gather their resources; break the problem down into smaller, achievable steps; and solve their problem by fast prototyping and incremental development. In a course with interdisciplinary collaboration, new challenges emerge, including the difficulty in achieving integration and balance in the contributions from the two disciplines. One of the big challenges is that students may want to work on something that you're not very familiar with yourself. In open-ended problem solving, you don't necessarily know what's down some of those paths, so it makes for daring teaching. Often, some of the knowledge needed for a project turns out to be outside of or tangential to computer science. For example, hands-on learning these days often involves electronics – Arduino boards, Raspberry Pis, Bluetooth devices, etc. And get out your soldering iron!

What I'm advocating is a move toward not just "need to know" learning, but "want to know" learning for students and "surreptitious teaching" for educators. The teaching in project-based courses requires that the instructor stand to the side, lying in wait with relevant pieces of fundamental scientific knowledge, mathematical analysis, or artistic perspective to deepen the students' hand-on work; and ready to move down unexpected paths that require the instructor to learn along with the student. Imagine the questions students might pose and explore: What makes a fractal a recursively-defined structure? How would I draw one by hand? How hard is it to compute a fractal and display it pixel-by-pixel, and what does it mean, to "zoom in" on one? How much time does the computation take? How fast does music play? How do you figure out the frequencies in music, and how much time does that take? Do flexible display surfaces exist? How would another surface detect the prominent features on my face? What kind of computer program could warp my facial features in a funny or scary way? What's a convolution? How would I communicate with the program? What hardware would I need for my morphing mask?

As we move toward the future, we STEM educators are called upon to help our students imagine what useful, creative, life-affirming things they can do with their digital devices. As we engage students in hands-on projects, we discover that active learning is as active for the teachers as for the students. Allowing students the freedom to follow their own imaginations requires thoughtful deliberate, purposeful, courageous teaching. In all, we're setting the bar high for ourselves as we remain mindful of the great influence we can have on the problems students choose to solve in this world, and the ingenuous, original ways they might devise to solve them.

References

Burg, J., et al. "Engaging Non-Traditional Students in Computer Science through Socially-Inspired Learning and Sustained Mentoring." *Proceedings of SIGCSE 2015*, March 2015, Kansas City, MO.

Biographical Information
Jennifer Burg is a professor of Computer Science at Wake Forest University. Her work focuses on the physical, mathematical, and computational elements of digital 2D images, sound, and music, and related pedagogy. She has written two textbooks on the subject: *The Science of Digital Media* (Prentice-Hall 2008); and, with co-authors Jason Romney and Eric Schwartz, *Digital Sound and Music: Concepts, Applications, and Science* (Franklin Beedle 2016).