Achieving Gender Balance through Creative Expression

William H. Bares, Bill Manaris, Renée McCauley, Christine Moore
College of Charleston, SC USA
Overview

- Gender balance can be achieved by providing opportunities for *creative expression throughout* the four years of an undergraduate computing degree.
- We present a four-course sequence combining computing and the arts that can serve as the core of a Computing in the Arts major or a concentration to *complement* an existing Computer Science major.
- A robust *six-year longitudinal study* shows that these courses attract, retain, and graduate 46% *female students*, while the corresponding number in our ABET-accredited Computer Science curriculum is approximately 20%.
- Concludes with observations and suggested directions for future action in CS curricula design.
The national graduation rate of female students in computer science is 17.9% according to the Taulbee Survey.  

The national graduation rate of female students in computer science is 22.1% according to the ACM-NDC Survey.
Foundations: Contextualized computing

- Contextualized computing focuses on applying computation in a particular domain of interest.
- Guzdial posits contextualized computing as a method of retention:

  A context provides relevance and improves retention. Students are more likely to stick around if they understand the value of what they’re learning. There is no evidence that they learn the material better, but they’re more likely to be around in the second class, even if that’s a normal decontextualized course. [p. 6] 17

Foundations: Media Computing

- Guzdial’s work inspired a large community of educators to adopt pedagogical materials introducing computer programming in the context of the arts.

- Examples:
  - Lewandowski et al. offer open-ended, hands-on opportunities for creativity in 3 milestone classes.
  - Uludag et al. use complementary technologies (Scratch, App Inventor) to increase appeal to broad audiences and attract both non-majors and majors.
Benefits of Media Computation

- Overall, media computing courses have demonstrated lower failure rates compared to traditional computer science courses [12, 26, 29].

- Media computation courses have motivated greater numbers of students to take additional media computation courses [13, 29, 30].
Creativity

- Plucker et al. 2004 define creativity as “the interaction among aptitude, process, and environment by which an individual or group produces a perceptible product that is both novel and useful as defined within a social context.”

- Expressions of creativity are individualized “novel and personally meaningful interpretation of experiences, actions, and events.”


Gender Balance and Creativity

- Courses integrating creative expression with computer science are often more gender-balanced.

- The Georgia Tech’s Bachelor’s degree in Computational Media, is reported to have 45.3% female enrollment as compared to only 17.91% female enrollment in the computer science major. ¹⁹

- Barker [5] reported that the Technology, Arts, & Media Bachelor’s degree [6] at the University of Colorado, Boulder has a 44.8% female enrollment.

At College of Charleston, we created a course that combined Python, fractals, L-Systems, and jMusic (a Java-based system for composing music with computers).

This course was initially called “Media Computation”, we soon renamed it “Computers, Music, and Art” (Fall 2007).

Supported by NSF DUE (1323605 and 1044861).
CITA Synthesis Courses

- The synthesis courses, which contextualize art and computing, are vertically integrated throughout our CITA curriculum, and thus provide opportunities for **creative expression throughout the four-year undergraduate experience**.

- Course assignments encourage creativity while maintaining rigor by outlining required computing and artistic outcomes.

- For example, we specify required algorithm, data structure, and user interface features while providing the freedom to creatively design the artistic content (visuals, game, music, etc.).
CITA Synthesis Courses

- Synthesis 1: basic programming concepts situated in an accessible and creative arts-themed context.
- Synthesis 2: applying software development in the context of game programming.
- Synthesis 3 - Seminar: Students develop a proposal for a capstone project, which synthesizes creativity in the arts with computing (pre-req: three classes in chosen art area).
- Synthesis 4 - Capstone: In the fourth-year capstone practicum, students implement their projects according to their proposal plans.
Additionally, CITA majors complete 9 credits in upper-division CSCI coursework, and 18 credits in their art concentration area.
Enrollments in synthesis courses (2011-2017)

<table>
<thead>
<tr>
<th>Course</th>
<th>F</th>
<th>M</th>
<th>F%</th>
<th>Course</th>
<th>F</th>
<th>M</th>
<th>F%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synth1</td>
<td>224</td>
<td>384</td>
<td>36.8</td>
<td>CS1</td>
<td>350</td>
<td>860</td>
<td>28.9</td>
</tr>
<tr>
<td>Synth2</td>
<td>50</td>
<td>107</td>
<td>31.9</td>
<td>CS2</td>
<td>217</td>
<td>650</td>
<td>25.0</td>
</tr>
<tr>
<td>Seminar</td>
<td>24</td>
<td>37</td>
<td>39.4</td>
<td>Software Architecture</td>
<td>56</td>
<td>241</td>
<td>18.9</td>
</tr>
<tr>
<td>Capstone</td>
<td>21</td>
<td>27</td>
<td>43.8</td>
<td>Practicum</td>
<td>45</td>
<td>199</td>
<td>18.4</td>
</tr>
</tbody>
</table>
Synthesis 1: Learning Outcomes

- Apply numeric and string data types to represent information.
- Apply variables in program development.
- Understand arithmetic operators and apply them to design expressions.
- Understand loops and use them to design processes involving repetition.
- Understand and apply processes involving selection.
- Understand functions and use them to design processes involving modularization.
- Use predefined classes in program development.
- Learn basic principles for group collaboration.
Computers, Music and Art

- Example Assignment: Create a sonification of an interesting image.
- Samples of student work:

See https://vimeo.com/64109534

See https://vimeo.com/64101616
Computers, Music and Art

- Example Assignment: Create a sonification of an interesting image

- Computing learning outcomes:
  - Apply a 2D list data structure to store image data (RGB pixels).
  - Apply a nested count-controlled loop to iterate over the pixels.
  - Apply a mapping strategy to convert data from one domain (pixels) to another (notes).

- Artistic learning outcomes:
  - Compose an original piece by selecting a beautiful or compelling image and then designing a set of musical parameters to convert this image to sound. Musical parameters include pitch and scale, dynamics, timbre, and instrumentation.
Animation & Virtual Worlds

- Example Assignment: Create a particle system animated 3D scene.
- Samples of student work:

  - Fireflies over a moonlit lake
  - Helicopter chasing an Audi in the rain
Example Assignment: Create a particle system animated 3D scene

Computing learning outcomes:
- Apply a list data structure to represent particles each having at least a position \((x,y,z)\) and velocity. You may add extra attributes for color, lifetime, size, etc.
- Apply a count-controlled loop to iterate over the particles.
- Apply a Boolean stopping condition to determine when a particle has expired and needs to be either re-generated or removed from the list.
- Apply keyframe animated properties to visualize each particle
- Apply basic physics equations to update particle position and velocity

Artistic learning outcomes:
- Compose an original 3D scene by arranging existing 3D models to create a backdrop that fits the visuals of your particle effect. For example, create fireworks over a cityscape.
Digital Media Programming

- Example Assignment: Design an interactive art composition or a game.
- Samples of student work:

![Space Snake](image1)

![Mama Earth, Female Ruler of the Galaxy](image2)
Digital Media Programming

Computing learning outcomes:
- Use **Object Oriented Programming** to include at least one class.
- Incorporate various **data types**, including arrays and text.
- Comment liberally to properly **document** the program.
- Apply variables, **conditionals**, and/or **loops** to create non-static objects.

Artistic learning outcomes:
- Most graphics must be **created from code**, although you may use a limited amount of raster images.
- Apply **principles of graphic design** to result in an aesthetically pleasing interface.
Synthesis 2: Game Programming

- **Design Learning Outcomes:**
  - Apply an iterative creative design process including *prototypes* (paper, digital, and code), gathering player *feedback*, and *revision*.
  - Apply *game design principles* to model player experience goals, balance challenge versus engagement, design sprite artwork.
  - Apply *user interface design principles* to provide feedback.

- **Computing Learning Outcomes:**
  - Apply an *iterative* software design process (testing plan, design, development).
  - Apply *lists* of objects to animate objects and check collisions.
  - Apply *finite-state machines* to control behavior of game objects.
  - Apply *maze navigation* algorithms in a 2D tile-based map.
Synthesis 2: Game Programming

- The course culminates with a team game-development project exercising all course outcomes, which is presented and play-tested in the final class meeting.

- We chose to use Processing, since it bridges the Python and Java languages used in our CS1 and CS2 courses.

Sample game:
2D tile grid
Randomly placed objects
Bird diving moves
Cat jumps and lands on platforms
Seminar: Learning Outcomes

- Explore archived examples of previous CITA capstone projects.
- Conduct literature searches to identify prior works that blend art and computing.
- Apply iterative software development through prototyping, revision, and testing.
- Produce a working code prototype which demonstrates a solution for a critical technical element of the proposed project.
- Written proposal:
  - how you synthesize art & computing
  - expected outputs and behaviors
  - algorithm design and justification
  - task timeline
Seminar

- Students must interact with faculty **mentors** in art and computing, who offer early feedback on artistic and computing aspects of the students’ project ideas.

- These early contacts help students to (a) create projects that better synthesize art and computing, (b) identify and correct problems early on, and (c) help students to change directions when necessary.

---

**Palette Jack**

Interface with MS Kinect in C++

Compute color palette of image

Dynamic 2D visuals animated using computed color palettes
Capstone: Learning Outcomes

- Present work-in-progress demos of milestones to peers.
- Consult with one art mentor and one computing mentor to solicit feedback on progress.
- Offer constructive feedback to third-year synthesis students as they work through the process of identifying their projects.
- Make a formal presentation of the completed project.
- Write report summarizing technical details of the solution and reflecting upon the completed work.
Observations and Lessons Learned

- Even if one cannot implement a whole curriculum, starting small – even at the level of a single course – has a positive impact.

- Our courses “Computers, Music, and Arts” and “Game Programming” were offered before our CITA degree was designed (Fall 2006).

- We observed that these courses attracted many non-majors and women, encouraging us to create the CITA degree.
Observations and Lessons Learned

- It could also be argued that the consistently high percentage of female students through all four-years of the CITA synthesis courses supplies the necessary critical mass and role models which contribute to increased self-efficacy due to vicarious experience (Bandura, 1986; Bandura, 1997).

- Introducing opportunities for creative expression anywhere within a traditional Computer Science curriculum could help.
  - For example, the concept of lists of objects may be taught in a typical CS2 or Data Structures course by implementing insert, search, and delete operations for student records.
  - To offer greater opportunity for creative expression, lists of objects could be used to create an animated particle system effect.
Conclusions

- A six-year longitudinal study demonstrates that combining creative expression and the arts with core computer science courses (including computer programming, data structures and software engineering) results in graduating 46% female students (compared to the 20% traditional nationwide statistic).

- In our experience, 72% of CITA graduates are placed in CITA-related jobs or graduate studies that involve computing or arts, with a majority of them in jobs that require their computing skills.

- A CITA-like degree is a viable complement to a traditional Computer Science degree, as it prepares students for careers in computing, while at the same time eliminating the gender disparity found in traditional computing degrees.
Supplemental Slides (for Questions)

- Data collection methodology & limitations
- CITA attracts new/different students to computing
- College of Charleston institutional profile
- Placements of CITA graduates
Data Collection Methodology

- Office of Institutional Research at our institution provided:
  - final grade records from Fall 2011 to Spring 2017
  - Data included course name, course number, section, final student grade, student gender, student race, and student major (or majors, if more than one).
- Our institution’s student records system stores only the latest (current, or final) student major as of Spring 2017.
  - Unable to track when students changed majors.
- We wrote various Python scripts to clean up and compute the statistics reported.
Synthesis 1 Courses Attract New Students to Computing

Top Seven Non-Computing Majors by percentage enrolled in Synthesis 1

- Business (22%, or 53 of 239)
- Arts (22%, or 52 of 239)
- Lab Sciences-Math (16%, or 39 of 239)
- Social Sciences (10%, or 23 of 239)
- Humanities (4%, or 11 of 239)
- Education (3%, or 9 of 239)
- Languages (1%, or 3 of 239)

Top Seven Non-Computing Majors by percentage enrolled in Computer Science 1

- Lab Sciences-Math (49%, or 202 of 450)
- Business (17%, or 75 of 450)
- Social Sciences (6%, or 25 of 450)
- Arts (5%, or 22 of 450)
- Humanities (3%, or 14 of 450)
- Languages (2%, or 8 of 450)
- Education (1% or 4 of 450)
College of Charleston

- State institution, primarily undergraduate student population
- Focus on faculty teaching undergraduates, little to no use of TAs
- 10,375 undergraduate students
- 64.4% are women
- 19.1% are from underrepresented populations as of the first semester of the 2016 academic year
Placements of CITA Graduates

- 72% of CITA graduates are placed in CITA-related jobs or graduate studies that involve computing or arts, with a majority of them in jobs that require their computing skills
- digital collections coordinator at College of Charleston Halsey Center for the Arts
- performing musician and works for non-profits who advocate for social justice
- user interface design and testing
- support specialist
- website designer and programmer
- graduate school (Clemson Digital Production Arts, MFA)