Redesigning Creative Coding Education for Beginner Programmers

MUSE Students: Thendral Prabu & Eliana Gargiulo
Faculty Mentor: Joshua Fishburn

TCNJ THE COLLEGE OF NEW JERSEY

Summary

The intent for this project was to improve the learning experience of first-time programmers by designing a learner-centered online resource for creative coding (applied programming in the arts) in p5.js. While there are established introductory texts for Computer Science, the equivalent texts for creative coding are not as well defined.

Our team hypothesized that beginner programmers, especially those from groups what are traditionally underrepresented in computing fields, would benefit from an open educational resource that takes a learner-centric, project-forward approach to teaching computing concepts through creative coding languages.

Overall, we would summarize our methodology as exploratory, interdisciplinary design research.

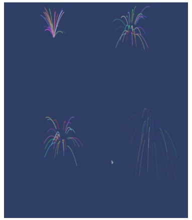
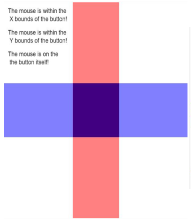


FIGURE 1: DYNAMIC FIREWORK PROJECT EXAMPLE



The mouse is within the X bounds of the button!

The mouse is within the Y bounds of the button!

The mouse is on the the button itself!

FIGURE 2: RECTANGLE COLLISION VISUALIZATION

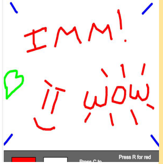
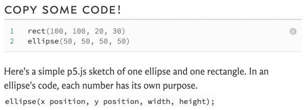


FIGURE 3: DRAWING TOOL PROJECT EXAMPLE



COPY SOME CODE!

```
rect(100, 100, 20, 30)
ellipse(50, 50, 50, 50)
```

Here's a simple p5.js sketch of one ellipse and one rectangle. In an ellipse's code, each number has its own purpose.

`ellipse(x position, y position, width, height);`

FIGURE 4: COPYABLE CODE SNIPPET PROTOTYPE

Process

Platforms: Roam Research, MindMeister

Teaching Language: p5.js

We analyzed the content of the textbooks and academic writing alongside our own experiences with computing education to develop a set of criteria to identify the successful components of an educational resource.

The criteria are as follows.

- Coordination of color, graphics, and text for an engaging read
- Access to community of like-minded learners, teachers, and professionals
- Use of simple, straightforward language
- Establishing relevance to the learner through historical blurbs and connections to careers
- Publisher credentials, author's experience, classroom evaluations/student and teacher feedback
- Content organization and structure

We established that there is significant evidence of student success when applying a learner-oriented approach to computing education. Outstanding resources establish a connection to the learner's interests and motivations and employ interactivity in their graphical and programming elements to increase engagement.

To design, lay out, and host our project, we each developed sample lessons using Pollen, Observable.io, and HTML/CSS authoring environments to experiment with teaching styles, content organization and visual presentation.

Research

We analyzed the strengths and weaknesses of existing creative coding resources, such as the book *Getting Started with p5.js*, and lessons on the website *Khan Academy*.

Researching the basics of teaching computational, we discovered Allison Elliott Tew's "10 Constructs", and explored the concept of the notional machine. We also looked into research on the efficacy of traditional CS education, which informed where our work would diverge.

Foundational to our later work was the Eccles Model, which examines how and why learners make the academic choices they do. It highlights the criteria that students take into account when making decisions, like Cultural Milieu and Expectation of Success.

We reviewed a number of programming textbooks and resources, noting the differences between traditional computer science and creative coding textbooks to understand student motivation and engagement.

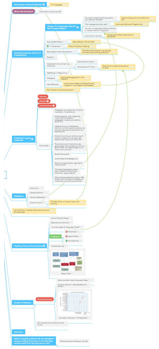


FIGURE 5: SNAPSHOT OF TOPIC MAP USED TO ORGANIZE RESEARCH

Design Philosophy

Based on our conclusions, we developed a design philosophy for our resource.

Our resource would be organized by project, not by topic, to better appeal to our audience.

We would make regular use of graphics and live examples to help demonstrate programming concepts.

These decisions led us to choose the language Pollen as the platform for our resource, since it supports both our organizational scheme and plans to include interactive elements.

Outcomes & Next Steps

At the conclusion of our research, we had a definitive project-centric chapter outline around which we designed multiple projects for students to exercise skills taught in each chapter and emphasize skills built in previous material.

One full chapter, 'Exploring Input in p5.js', was fleshed out according to the principles and research established earlier in our work with emphasis on including interactive graphics.

In future research/projects/studies, work will be focused on building the book's platform and the tools inside it in Pollen, HTML and CSS and on producing a cohesive typography and color scheme in which the pedagogical content would be framed.