Techniques at the Intersection of Computing and Music

Jesse M. Heines
Dept. of Computer Science
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-3634
heines@cs.uml.edu

Gena R. Greher
Dept. of Music
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-3893
Gena_Greher@uml.edu

S. Alex Ruthmann
Dept. of Music
Univ. of Massachusetts Lowell
Lowell, MA 01854
1-978-934-3879
Alex_Ruthmann@uml.edu

Abstract

Our work on *Performamatics* aims to enhance students' computational thinking (CT) by engaging them in fundamental concepts that unite computing and music. Our approach leverages students' near universal interest in music as a context for rich CT experiences. The techniques we share are used in a General Education course open to students in any major called *Sound Thinking*, which is now being offered for the fourth time.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *computer science education, curriculum.*

Keywords

Performamatics, computer science education, interdisciplinary programs, music+computing, Audacity, Scratch.

Course Structure

Our course teaches both music and computing. We begin by having students analyze music through listening exercises and sequencing and manipulating sounds using Audacity (audacity.sourceforge.net). We then have them begin to work with prerecorded sounds in Scratch (scratch.mit.edu) before getting into composing in Scratch with MIDI.

Audacity-Based Techniques

Students begin the course by making instruments from "found objects," that is, common things that they find in their living area. We specify that they should look for objects that can produce multiple pitches with multiple timbres. This gets them thinking about what, exactly, constitutes "music."

The next step is to record the sounds their instruments make and load them into Audacity. They learn to use the editing capabilities of that program to break their sounds into component elements and the various effects to manipulate them. They then arrange the component sounds and apply effects to create a composition that further explores their own interpretation of "music."

In the third assignment, students choose a song of their own liking and listen to it critically to identify the components from which it is built. With these in hand, they create a flowchart of the song to illustrate how those components go together.

Scratch-Based Techniques

We then get into programing with Scratch, which is very easy for students of all majors to pick up and has some terrific capabilities for working with music.

Students begin by sequencing sounds in Scratch in a manner similar to what they did in Audacity. There is much more they can do, however, using loops and control structures. They also learn about the limitations of computer software as they try to do things that the system just can't handle.

Next we introduce intervals, focusing on the major 2nd and the perfect 5th. We show students how to program these and we have them create a composition using only these intervals. More advanced students program the intervals algorithmically and add randomization to have the music change each time it is played.

To introduce lists, we have students work on a piece that they must transpose from one key to another. This forces them to use offsets controlled by variables.

Finally, we introduce physical interaction between Scratch and the IchiBoard, a small device packed with sensors, a slider, a button, and accelerometers from which Scratch can read values that control the notes it plays (see www.cs.uml.edu/ecg/index.php/IchiBoard/IchiBoard). This technique gets students doing the most sophisticated programming in the course, as they must scale the sensor values to ranges that are appropriate for playing music.

Culminating Performances

Our course builds to a "recital" in which students showcase final projects based on the course assignments. Performances are in a venue other than our normal classroom, and students are welcome to invite their friends and family. Middle and secondary school students also attend, not only for their own benefit, but for the benefit of our students, as well. By explaining their work to younger students, our students expand and deepen their own understanding of the intersection between computing and music.

Our presentation will feature videos, recordings, and live examples of student work to demonstrate the types of results that students produce for the assignments described above.

Project URLs and Locations of Materials

Further information on our work and the materials that we use in our classes may be found at:

- *Performamatics:* http://www.performamatics.org
- *Sound Thinking:* http://soundthinking.uml.edu

Acknowledgments

This work is currently supported by National Science Foundation Award No. 1118435, "Computational Thinking through Computing and Music." *Principal Investigator:* Jesse M. Heines. *Co-Principal Investigators:* Gena Greher and S. Alex Ruthmann. *Senior Personnel:* Fred G. Martin and Sarah Kuhn. Any opinions, findings, and conclusions or recommendations expressed in these materials are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

ITiCSE'12, July 3–5, 2012, Haifa, Israel.
ACM 978-1-4503-1246-2/12/07.