Constrcuting Media as a Context for Teaching Computing and Motivating Women and Non-Majors

Mark Guzdial
College of Computing/GVU
Georgia Institute of Technology

1 Creating a Computing Course that Retains Students

The problem being addressed by this project is the disinterest in computer science exhibited by large groups of students, especially non-CS-majors and women—a particular problem at institutions like Georgia Tech where an introductory computing course is required.

• Percentages of women and minorities are dropping in Computer Science.

• The problem being addressed by this project is the disinterest in computer science exhibited by large groups of students, especially non-CS-majors and women—a particular problem at institutions like Georgia Tech where an introductory computing course is required.

1.1 Example Programs

def clearRed(pixel):
    for pixel in getPixels(picture):
        setRed(pixel,0)

def grayScale(picture):
    for p in getPixels(picture):
        intensity = (getRed(p) + getGreen(p) + getBlue(p))/3
        setColor(p, makeColor(intensity, intensity, intensity))

1.2 Syllabus

Our syllabus is based on an observation from the learning sciences that students can’t learn abstractions on experience until they have concrete experiences. These students have little to no previous programming experience. Our goal in the first 2/3 of the course is to motivate them to develop concrete programming experience.

• Sound as a media type, including psychophysics (how human hearing limitations make MP3 compression possible), looping to manipulate volume, then indexing by index numbers to do splicing and reverting of sounds.

• Text as a media type: Searching for text, composing text, reading text from a file and writing it to a file.

• Shifting media representations, e.g., creating visualizations of sound.

• Movies: How persistence of vision makes animations and movies possible, generating frames using the various techniques described earlier in the semester, manipulating whole directories of files.

Assignments are often open-ended. For example, we ask students in the third week to build a collage where images appear multiple times with some image manipulation, but students can choose their images and the manipulations (Figure 2). Similarly, we ask students to build audio collages and movies. Students are invited to post their created media in a shared CoWeb (Collaborative Website) for all to see (http://coweb.cc.gatech.edu/~cxl135). Students then have real questions that Computer Science can help them answer.

• “Can’t we do this any faster? Why is Photoshop faster than Python?” Introduction to how a computer works (e.g., machine language), and the difference between an interpreter and a compiler.

• Algorithmic complexity and the limits of computation.

• “Can we do this any easier?” Decomposing functions, modularly, and functional programming (map, reduce, fold, and simple recursion). Introduction to objects and classes.

• “What do other programming languages look like?” Brief overview of JavaScript and Squeak.

1.3 Course Curriculum

We teach Python (http://www.python.org, specifically the Java-based variant Jython http://www.jython.org)

def clearRed(pixel):
    for pixel in getPixels(picture):
        setRed(pixel,0)

def grayScale(picture):
    for p in getPixels(picture):
        intensity = (getRed(p) + getGreen(p) + getBlue(p))/3
        setColor(p, makeColor(intensity, intensity, intensity))

1.4 Data Sources and Methods

When asked what they like about the class in the midterm survey, the students affirmed that we’re succeeding at creating a course that students recognize for its relevance, particularly for non-CS majors.

• “Very applicable to everyday life.”

• “I dreaded CS, but ALL of the topics thus far have been applicable to my future career (and personal plans).”

• “The professor answers questions in class and online and is concerned about our success in the class. He also seems to understand that most of us are not engineers and most likely won’t do straight programming in the future—just the way of thinking is important.”

• “I think that we’re doing things that I could actually use as an architecture major-I like dealing with pictures and sounds.”

5 Next Steps

• The course has been adapted by several schools, in entirety (e.g., Gainesville College) or parts (e.g., Brandeis University, University of Maryland at College Park, DePauw University).

• Some of our students have become CS majors. In response to their interest in more Computing, we have defined a second media computation course, Representations of Structure and Behavior, which will cover data structures content. The two courses will form a pathway into our CS major, newly defined CS minor. With our New Media Center in the School of Literature, Communications, and Culture, we are defining a BS in Computational Media.

• We have been asked by the Georgia Department of Education to use our media computation approach to teach high school teachers how to program so as to increase the number of certified computing and CS AP teachers in the state. We are adapting our materials to Java for this purpose.

Funded in part by a National Science Foundation CCLI “Proof of Concept” grant, GVU Center and College of Computing, and the AI West Fund at Georgia Tech. Evaluation effort led by Andrea Forte with Lauren Rich, Rachel Fithian, and Heather Perry. Software developed by teams of (mostly undergraduate) students led by Adam West and David Raines.