# Commentary: Week 4

CSE120: Computer Science: Principles

This week the goal is to promote creativity with programming. We also solidify our understanding of the Processing Visualization Language by covering a series of more ambitious facilities.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lecture 9 | Lab HW work | Lecture 10 | Lab HW work | Lecture 11 |
| Assignment 9/10 | | | | |

**Lecture 9:** This lecture focuses on the controlling input/output with the mouse, keys and text. Each topic is covered. Each topic would benefit from another example, because students seem to watch the lecture, but not really grok the idea. Happily, the creativity of this week’s assignments motivates them to return to the slides and learn it.

****

**Assignment 9/10:** This is a combined homework assignment where students produce four small apps during the week that do something that would interest a viewer for a few moments. Very simple examples are presented, but most students have at least one or two ideas for something very cool. Perhaps the all- time best one from last year is the “Magic 8 Ball” which vibrates around like it is being shaken; when a key is pressed, the “8” progressively disappears and a blue triangle with the “answer” slowly appears. It is very cute, and included in the files.

**Lab HW work:** The labs this week are devoted to giving students an opportunity to work on their creativity assignments with “assistance”. They often are very ambitious, and need either to be reined in, or helped to move forward.

**Lecture 10:** The topic is information. It’s a fun lecture. The idea of a bit is given, and for a while it is used without converting to 0/1. Eventually, the shear burden of not having a short name motivates us to pick these to words, but we emphasize that they are only symbols, not numbers. We also encode bits, discuss compression Huffman-like encoding. We do ASCII and UTF-8.

**Assignment 9/10:** See above

**Lab 8 HW work:** Normally, we teach counting in binary, binary representation, conversion to/from decimal and addition in this lab. (It sounds like a lot, but much of this moves quickly.) My students were asked to watch a couple of videos to save me from going over binary and make the time available for homework questions. They didn’t watch them. Ultimately I had to go over this material on several occasions. I still think this is the most efficient way to teach this, but I returned to teaching it manually this time.

The video resources were:

A good vid on binary: https://www.youtube.com/watch?v=ry1hpm1GXVI

A quick tutorial on binary addition: http://courses.cs.vt.edu/~csonline/NumberSystems/Lessons/AddingTwoBinaryNumbers/index.html

A slick self test, with answers on addition: http://courses.cs.vt.edu/~csonline/NumberSystems/Questions/AddingTwoBinaryNumbers/index.html

**Lecture 11:** This lecture’s topic is the key idea – bits are good enough! They are universal and encode ALL information. But first … The lecture begins with pixels. We discuss how the screen is designed and implemented. Also, we discuss why RGB, using light, isn’t like the color information everyone learns in elementary school, which uses reflectance. It answers why yellow is R+G. We do analog sound, compression of GIF and JPG, and finally land on the main principle.

**Assignment 11:** This is a new assignment this year and I believe quite successful. Its goal is to introduce arrays and to create a fun activity. (Later assignments indicate students go the content pretty well.) The idea is to create a caterpillar out of seven segments. (A version of the code elli.txt that makes a random walk, and so is more complicated than needed for the assignment, is included.) We will use this in the next lab, making it controllable. The assignment steps through using the worm code as illustration of the array concepts.