Introduction to CSCI 1311

Dr. Mark C. Lewis
1-13-2010

Opening Discussion

- I will start off every class with a little discussion. Typically this discussion includes questions about the previous class.
- Today I want to use this time to get to know a bit about you.
- I'd also like to give you a brief introduction to me so you can know some of the things that motivate me.

Course Basics

- Course web site: http://www.cs.trinity.edu/~mlewis/CSCI1311-S10/
- Office: HAS 201K
- Office hours: 2:00-4:30pm MW, 1:00-3:00pm T, and 1:00-4:30pm R or by appointment. The afternoon hours are open labs.
- Phone: 7022
- E-mail: mlewis@trinity.edu (This is the best way to reach me most of the time.)
- There is a schedule on the web site listing all topics and when things are due. There are links to lecture notes.

Text

- I am working on the text for this class. The links page has a link to the PDF file. It is a work in progress and will be changing over the course of the semester.
- Mostly it should be getting longer as I add some chapters.

Course Description

- This course provides an introduction to programming with a primary focus on problem solving and logic.
- The course will specifically teach you how to program in Java. We will be using different environments for Java programming. Everything can be done in Windows.
- My courses tend to be somewhat rigorous. My overriding objective is to make you think. Because of the nature of this course I want to go beyond just giving you new things to think about, but instead give you tools to think about things in new ways.

Creativity of CS

- Writing great programs is more art than science.
- Computers are the ultimate creative medium and also the ultimate creative outlet. The way we interact with them seems formal to some, but there can be a lot of fun in just looking for ideal ways to express ideas in the formal systems of different computer languages.
- I will try to let you express creativity in the work you do for this class.

Projects

- The largest chunk of your grade in this class is determined by the two projects that you will do.
- These projects will be fairly large scale where you apply all of the elements that we have learned about in each language.
- You can work with others on design and get help on specific problems, but what you turn in must be your own construction. Everything you turn in for a grade is pledged under the honor code.

Grading

- Your grade comes from four different components.
  - Projects (2) – 40%
  - Tests (2) – 30%
  - Quizzes (6 with lowest dropped) – 10%
  - Interclass Problems – 10%
  - Class Participation – 10%
- The midterm and final are both equally weighted.
- The quiz questions are modeled after test questions so you will have an idea of the style to expect on the test.
- Interclass problems are presented about once each week.
- Class participation includes attendance and your actual participation during class.

Interclass Problems

- I use interclass problems to help you determine how much you have to read to master each topic, and to give you programming experience. Each of you will be called on a total of five times with each time being worth two points to your final average.
- If you don't have anything to show you get zero points. Complete and working answers get two points. Limited efforts and incomplete results will give you one point.

What is a Program?

- A program is a set of instructions we give to a computer to make it do something.
- There are many languages we can do this in.
- A general commonality is that the instructions must be explicit and precise. Programming languages do not allow ambiguity.
- This is what makes programming so helpful for all people to learn.

Why Should You Care?

- Programming is an obvious skill for the computer scientists. Why does it matter to other people?
  - Logic and problem solving
  - Tedious/repetitive tasks
  - Number crunching: Excel macros
  - Communicating with technical people
  - Understanding of technology

Environment for this Class

- We will be using Windows for all of the instruction in this class.
- We will start off using Greenfoot. Let's go ahead and open it up and see how it works.
- Later in the semester we will add the Eclipse environment. We won't really worry about that until we get there.
- Things should work fairly well on a Mac as well.

Basics of Greenfoot

- Let's open up Greenfoot on these machines and look at it briefly.
- Greenfoot is an environment that helps you learn programming by attaching a 2-D environment to everything that you do.
- Java is an “object-oriented” language. The graphical environment helps to make abstract concepts more concrete.

Minute Essay

- At the end of each class I ask one or more questions and you are supposed to answer them on a piece of paper. Make sure it has you name as this is how I do attendance.
- Do you have any questions about the course or what we will be doing?
- You should install Greenfoot on your computer. Load in each of the scenarios and play around some. Start reading the preface and chapter 1 of the text.