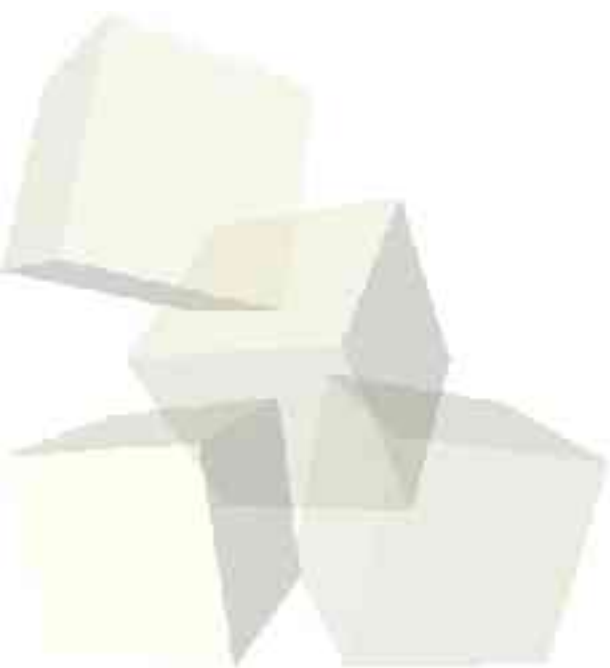




CSCI2322

Functional Programming

Dr. Mark C. Lewis
8-27-2004



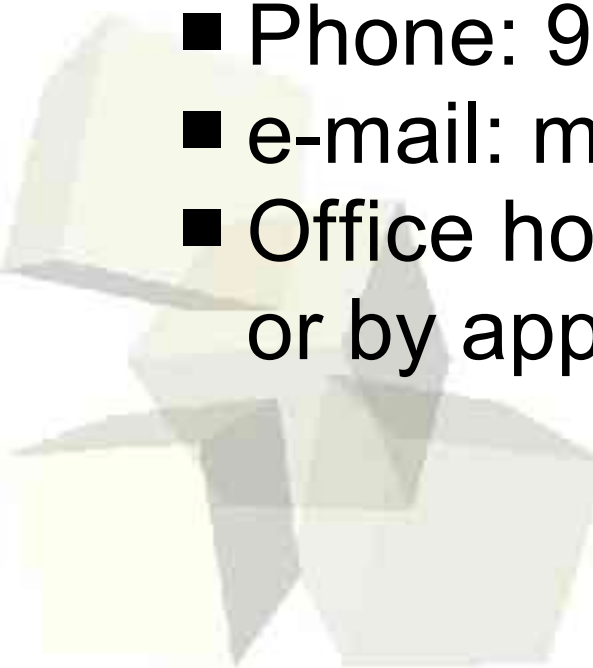


Opening Discussion

- I like to open classes with a discussion. In most cases I will ask you a questions about the previous lecture or for questions in general about what we are doing in class.
- Today is obviously a bit different because we haven't had a previous lecture. Instead, I'm interested in getting to know a bit about you and your background in this topic.
- Can anyone tell me what a functional language is? Why are we bothering to study them?



Course Basics

- Try to log onto the machines and bring up a browser.
 - Course web page:
<http://www.cs.trinity.edu/~mlewis/CSCI2322-F04>
 - Office: HAS 201K
 - Phone: 999-7022
 - e-mail: mlewis@trinity.edu
 - Office hours: 2:30-5:00 MF, 10:00-12:30 TR
or by appointment
- 



Course Description

- This course is intended to introduce you to the functional programming paradigm. PAD1 introduced you to imperative programming. PAD2 works on object-oriented programming. This class exposes you to yet a third different paradigm.
- This course can be taught as “PAD1 in a functional language”, but it is my goal to go beyond that and give you a greater feel for that paradigm as well as helping you to see the strengths and weaknesses of it.



Grades

- Your grade in this class will be determined for the combination of 4 components.

Assignments (10)	40%
Tests (2)	40%
Quizzes (6 drop 1)	10%
Class Participation	10%



Assignments

- The majority of the work you do for this class will be in the form of programming assignments. There will be 10 of these through the course of the semester.
- In many ways, the assignments will be similar to what you might have done in PAD1, especially in the early parts of the semester. Later in the semester we will diversify this some.
- The work you turn in for this must be your own. It is pledged for the first years.



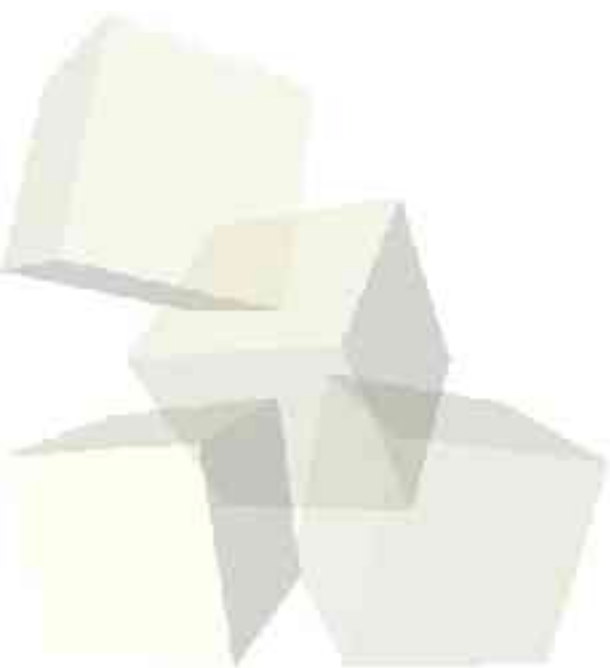
Schedule

- The most useful part of the course web site is the schedule. This lists readings and topics for every day of class. It also lists dates when assignments are due and when you will be taking quizzes.
- I also link PDF files with the lectures on this page so that you don't have to worry about writing down what is on the slides. Pay attention and think about what is being said instead.



Links

- The links page has a fairly large number of links that you can explore related to CS in general as well as the functional languages that we will be working with during the course of this semester.





Making you Think!

- In all of the classes that I teach, my overriding objective is to make you think. This course should help you to think about programming in very different ways than you have done before. Some of it might even bleed over into other CS courses that you are taking now.





What it means to be Functional

- Functional languages get their name from the fact that they are based on the mathematical concept of a function.
- The original functional language, LISP, was based on the Lambda calculus, which Church created and was known to be a Turing Complete model of computation.
- In a purely functional language, everything is based on functions. Given an input, it always produces the same output. Pure functional forms don't use assignments to variables.



Scheme Environments

- There are many environments that you can program in Scheme with. The simplest one, which I will originally use in class, is scm. Note that programming in Scheme is done in an interactive environment. If you like graphical interfaces, DrScheme is another alternative. We have both on these machines.
- Scheme environments also allow you to load in text files which is how you will be programming for this course.



ML

- In the second half of the semester we will branch out and do some programming in the ML programming language.
- ML is a more recent functional programming language that has some very interesting features to it.
- There are several derivatives of ML as well. One of which is O'Caml which mixes functional and object-oriented programming. Microsoft is working on a derivative of this called F# for .NET.



Minute Essay

- We will end each class with you writing me a short reply to some question. I use these to keep attendance, so make sure you put your name on them.
- In addition to/in place of answering the question, you can also give me any feedback you have or ask any questions of your own.
- If you had to decide now, what do you think we should do in the second half of the semester?