COSC 5340
Internet Programming
Summer 2004

INSTRUCTOR: DR. LAWRENCE OSBORNE
OFFICE: 201 MAES
OFFICE HOURS: 1:00 p.m. to 2:00 p.m. MTWR, and by appointment.

CLASS MAILING LIST: cs534005@lists.lamar.edu or

TEXTBOOKS: Sebesta, Robert W., Programming the World Wide


OTHER SOURCES: Deitel, Deitel, and Nieto, Internet & World Wide Web


Hall, Marty, and Brown, Larry, Core Web Programming, Second Edition, Sun Microsystems

Felton, Mark, CGI Internet Programming with C++ and

GRADING: Midterm Exam: 20 %
Final Exam: 30 %
Homework and Programming Assignments 50 %

A weighted average of your total points based on this grading scheme will determine
your final grade. Generally, the grading scale is based on cluster points in the
distribution of total points.
PREREQUISITES: Data Structures, C++ and UNIX, and a course in Operating Systems. A course in computer organization or computer architecture are helpful also.

GOALS

This course assumes that you are a serious software developer. This course is designed to provide the basics of HTML, XML, Java Applets, desktop applications in Java, Perl, CGI, Perl, servelets, PHP, and JavaScript. The underlying software at the transport layer, sockets, and RMI, is described and utilized in applications.

No one becomes a great developer just by reading or just by programming. You have to study the languages and protocols used, but you also must write some real code. In this course, students will do both. We will start with a few simple programs and give one more challenging exercise on each topic.

Course Policies

Cheating on an examination will result in a zero on the exam. Since your grade will be based on total points, a zero on either the midterm or the final will reduce your grade considerably. Academic dishonesty is an egregious offense against the entire class and will not be tolerated. The final examination will be comprehensive. If the midterm or the final is missed, a makeup exam will only be given in the case of a documented illness or death in the family. The fact that your car does not run or that you wish to be with your girlfriend/boyfriend at the hospital are examples of excuses that will not be accepted. Due dates for assignments will be enforced.

It is the responsibility of the student to find out what assignments have been missed after returning from an illness or other emergency. I feel no obligation to rescue students who do not turn in assignments on schedule. In fact, I could not do that even if I were so inclined.

All assignments should be submitted on a CD or floppy disk. All submissions should include the following:

All source code required to compile and run your program.
Usage instructions, in a regular text file, describing
How to compile your code
How to run your program
How to verify that the program is running successfully
A list of the classes used in your system, and a short description (a few sentences) describing the role of each in the system. This can be provided as Javadoc comments in the code itself, or as a separate text file.
The documentation required for each submission (usage instructions and class descriptions) will comprise 1/8 of the points of each assignment. The remaining points will be distributed amongst the parts of the assignment as specified in the assignment descriptions below.

Do not submit assignments that require a particular IDE or other commercial software to compile/run. For example, don't base your work on classes that come with Visual Cafe or Borland JBuilder, etc. We will not have this software available, and if we can't run your code, we'll have to subtract points. Also, the instructions on compiling and running your code must be independent of any IDE.

All assignments should be your own original work, not based on online examples/tutorials.

Late submissions will not be accepted. Students who feel that they require an extension must contact me at least 48 hours prior to the assignment due date to ask for an extension. Whether the extension is warranted, and the duration, is at the discretion of the instructor.

Topics Covered in Course

I. HTML
II. Cascading Style Sheets
III. Java Programming
   A. Java Syntax
   B. Graphics
      1. Applets
      2. Applications
      3. Drawing
   C. Mouse and Keyboard Events
   D. Multithreaded Programs
   E. Network Programming
      1. TCP and UDP
      2. Sockets
      3. RMI
IV. Server Side Programming
   A. HTML forms
   B. Java Servelets
   C. Using applets as servlet front ends
   D. Java Database Connectivity
   E. XML Processing with Java
   F. Perl
   G. CGI
   H. PHP
V. JavaScript
   A. JavaScript syntax
B. Customizing Web pages  
C. Handling Cookies  
D. Controlling Frames  
E. Integrating Java and JavaScript

Unfortunately, we may not have time to finish all of these topics, but we will try to cover the fundamentals of all of these subjects. It is your responsibility to learn what is in the reading assignments whether or not we are able to discuss every topic in class. Examinations will cover reading, lecture, and programming material.

Learning Outcomes of the Course

1. Facility with Programming in Java, Perl, Javascript, and PHP
2. Knowledge of the characteristics of TCP, UDP, and HTTP Protocols
3. Understanding of communications software in Java such as RMI, sockets, and threads
4. Ability to use HTML, Cascading Style Sheets, and XML to structure and display documents
5. Understanding of how to access databases through the WEB including the use of MySQL and JDBC
6. Ability to use Servlets with Web Servers
7. Understanding of CGI and its use with PERL
8. Knowledge of XML parsers, XSLT Style Sheets, and XML Schemas
9. Improved ability to communicate within a group in order to develop software
10. Improved ability to write technical information for computer science reports.
11. Improved ability to document software.
12. Encouragement towards lifetime learning in computer science.