Course: CS3490: Programming Languages

Description: Comparison of user and implementation aspects of several programming language paradigms, including logical, functional, object-oriented, and imperative. Topics include name binding, data objects, control constructs, subprograms, and concurrency. Projects will emphasize logic and functional programming, language translation and interpretation, concurrency and intermixing programming languages.

Instructor: Barry L. Kurtz
Office/Phone: 119 CAP Bldg., 828-262-7008
Office hours: MWF 11:00 – 11:50 and 2:00 – 3:30 or by appointment
Text: Programming Languages and Paradigms; J. Fenwick, B. Kurtz, C. Norris; this book is currently being written, you will receive printouts of chapters as they are covered in class

Grading Policy:
- Class Participation and Attendance: 5%
- Programming Exercises and Homework: 35%
- Project: 8%
- First and Second Exam (13% each): 26%
- Final Exam: 26%

These percentages represent guidelines and may vary during the semester. Examination grades will be curved when the exams are returned so that you will have a good indication of your relative class standing.

Exercises, Programs and Project
Hands-on programming exercises will occur during lecture. Except for Prolog, all programming will be in the Visual Studio environment. Program assignments will be more substantial than exercises and completed outside of class. In completing your course project you will intermix the use of four programming languages (C#, C/C++, VB, and F#).

Topics, Programming Languages, and Lectures

<table>
<thead>
<tr>
<th>Course Overview</th>
<th>Language Syntax &amp; Implementation</th>
<th>Imperative Paradigm</th>
<th>Functional paradigm - intro</th>
<th>Functional paradigm - advanced</th>
<th>Logic programming</th>
<th>Visual Basic and Excel Macros</th>
<th>Object-oriented programming</th>
<th>Concurrency</th>
<th>Project presentations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>C</td>
<td>F#</td>
<td>F#</td>
<td>Prolog</td>
<td>VB and Excel</td>
<td>C#</td>
<td>OpenMP &amp; MPI</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3 lectures</td>
<td>5 lectures plus exam</td>
<td>9 lectures plus exam</td>
<td>2 lectures</td>
<td>8 lectures</td>
<td>4 lectures</td>
<td>2 lectures</td>
</tr>
</tbody>
</table>
Teaching Philosophy
The textbook will be closely followed. Lectures will emphasize hands-on programming experiences and the theoretical foundations of the various programming paradigms. Exams will be based on lecture materials and programming activities. There will be a strong emphasis on being able to write program code in a variety of programming languages (Prolog, F#, C#, C/C++, VB).

Attendance and Classroom Participation
All students are expected to attend class unless absent with a valid, documented excuse, such as a note from the infirmary. You are allowed only three unexcused absences. In addition to lecture on the theoretical topics, class activities will focus on hands-on program development. Everyone is expected to participate in this activity on a regular basis.

Late Submission Policy
No exercises, programs, or other course components will be accepted late unless accompanied by a valid, documented excuse, such as a note from the infirmary.

Communications Policy
Your email account will be used to communicate detailed course information. You are required to check your email once a day during the school week.

Collaboration Policy

EXERCISES AND PROGRAMMING ASSIGNMENTS
Discussion of the assignment with the instructor is encouraged. Discussion of the assignment requirements in a natural language (e.g., English) with fellow students is allowed, but sharing code in any manner (files, printouts, screen images) is forbidden.

EXAMS
No discussion of any kind, except with the instructor, is allowed during exams. Access to books, notes or copying from a neighbor’s exam is strictly forbidden.