**Course:** CS5110 : Design and Analysis of Algorithms  
**Description:** Advanced data structures, advanced algorithms and selected topics  
**Instructor** Barry L. Kurtz  
**Office/Phone** 119 CAP Bldg., 828-262-7008  
**Office hours** MWF 9:15-10:15 and 2:00-3:00 or TuTh 10:00-12:00 or by appointment  

**Grading Policy:**

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework/Exercises</td>
<td>5%</td>
</tr>
<tr>
<td>Programs &amp; Analysis</td>
<td>30%</td>
</tr>
<tr>
<td>Presentations (two at 5% each)</td>
<td>10%</td>
</tr>
<tr>
<td>Class Project</td>
<td>10%</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>20%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>25%</td>
</tr>
</tbody>
</table>

These percentages represent guidelines and may vary during the semester. Letter grades will be assigned only at the end of the semester based on a class curve.

**Programming Assignments** All programming will be in C++ or Java and should compile on the cs system using g++ or javac (you can develop on any system). The emphasis in the programming assignments will be to compare the performance of alternative algorithms to solve a particular problem. You are encouraged to find all applicable source code in textbooks or on the Internet so that you don’t have to re-invent the wheel (but remember to credit your sources). You will be graded on having working programs, a systematic analysis of performance, and a brief write-up comparing and analyzing performance, as appropriate for each assignment.

**Class Presentations** Each student will complete two presentations on topics of his/her choice and approved by the instructor. Presentations will normally be theoretically oriented and not involve any new implementation. A complete report will consist of a short presentation in class plus submission of any display materials.

**Class Project** Each student will complete a class project of his/her choice and approved by the instructor. The topic should be related to the course materials but need not necessarily have been covered in class (for example, wavelets as an alternative to FFT). A complete course project will include implementation of new software, collection and analysis of data, as appropriate, and a class presentation.

**Course Objectives**

1. Appreciate the theoretical foundations for the analysis of algorithms  
2. Learn a wide variety of algorithmic approaches to various problems  
3. Write programs that measure the time complexity of algorithms, compare these with theoretical results, and analyze any difference between theory and practice
Teaching Philosophy  This course will follow the textbook very closely. Notes from the lecture will be handed out in class (funding permitting) and will be available on the instructor’s website. Exams will be based on lecture materials, exercises, and programming assignments. The Master’s qualifying exam in algorithms will be based on the same materials.

Schedule of Topics  A detailed list of topics is provided on a separate sheet.

Attendance Policy  
All students are expected to attend class unless absent with a valid, documented excuse, such as a note from the infirmary. Although not formally included in the percentages of the course grade, the instructor reserves the right to raise (e.g., B+ to A-) or lower (e.g., B+ to B) a student’s grade by one level based on class attendance and participation.

Program Submission Policy  
All programs will be submitted via the cs machine on the UNIX system using the following command:

```
/u/csd/blk/bin/submit5110  <program name>  <files>
```

The program names will be sequential: p1, p2, p3, and so forth. Remember multiple files can be submitted with one submission and wildcards are possible.

Late Submission Policy  
No programs, exercises, 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 on the “cs” machine will be used to communicate detailed course information. You are required to check your email once a day during the school week.

Collaboration Policy  
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 unless it is a group assignment, in which case you can share with group members.

EXAMS  
No discussion of any kind, except with the instructor, is allowed during exams. Access to books, notes or other material is strictly forbidden.