Fall 2004

CS 240: Introduction to Computer Science

Eric Matson

Wright State University - Main Campus

Follow this and additional works at: http://corescholar.libraries.wright.edu/cecs_syllabi

Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation


http://corescholar.libraries.wright.edu/cecs_syllabi/135

This Syllabus is brought to you for free and open access by the College of Engineering and Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact corescholar@www.libraries.wright.edu.
Computer Science (CS) 240  
Introduction to Computer Science  

Autumn Quarter 2004  
Wright State University  

Course Description  
We will develop basic techniques to design, develop and implement programs using the C++ language.  

Goal  
There are several goals to accomplish in CS 240:  
1. Master basic coding techniques and skills in C++  
2. Learn how to use an Integrated Development Environment (IDE) such as Visual C++  
3. Learn how to develop software programs  
4. Have some fun!  

Lecturer  
Eric Matson  
Office: 336 Russ Engineering Center  
Phone: 937-775-5108  
Office Hours: Tuesday/Thursday 1:00 - 2:15  
Email: matson@cs.wright.edu  
Web: www.cs.wright.edu/~matson  

Class  
• Monday/Wednesday 12:15 - 1:30 at Russ Engr. Center 153  
• Exams on 2 Fridays at Russ Engr. Center 153  
• Lab each week, starting in the second week of class  

Text  
Required: Big C++, Cay Horstmann and Timothy Budd (2005), John Wiley and Sons, Inc.  

Tools  
Microsoft Visual C++ 6.0 Compiler is available at the library.  

Required Work  
| Homework | 30% | 4 Programming Assignments  
| Lab | 20% | 8 Lab Assignments  
| Midterm Exams | 25% | 2  
| Final Exam | 25% | 1  

Grading  
The base scale is: A: 90-100, B: 80-89, C: 70-79, D: 60-69, F: 0-59. This is the highest requirement that will be used. The scales may be lowered or revised if necessary.
Policies and Notes

- Attendance: Attendance is not required, nor will it be taken after the first couple of lectures. If you are not a regular attendee, it will be your responsibility to seek out what material was covered in the lecture and learn it. Most of my exam questions will be taken directly from ideas covered during the lecture, so it greatly helps if you attend!

- I will utilize my CS web page (www.cs.wright.edu/~matson) to post updates to the course, sample code, projects, announcements, schedule, etc. Get in the habit of checking it regularly.

- There are no prerequisites for the course. Algebra and trigonometry are co-requisites for this class.

- Always make back ups of all of you work. Never have just one copy of anything!

- If you are going to miss an exam, for any reason, discuss it with me in advance. If it is an emergency situation, please notify me as soon as possible. *Sleeping late after attending the 24 hour Monty Python movie festival is not considered an emergency!*

- You can reach me a number of ways. Email is normally the best as I check it about 18 hours a day normally. You can also reach me by phone during the day at 775-5108. If you need human contact either stop in during my office hours, make an appointment, or just come by my office. If I am in and not on a deadline to get something else completed, I will normally try to help as much as possible.

- There are technologies we will use in this class that you may not already know, such as working with Integrated Development Environments (IDE). We will cover some of these technologies as we go or they will be discussed in lab. If you have trouble, please don’t hesitate to come and talk with me.

- The key to learning in this class will be spending time working through the problems. Don’t wait until 2 hours before something is due to try to learn the concept and then write the program. This normally ends in a disaster! Stay up with the readings and try to work through some of the examples in the book. I will post what I call, “10 minute programs” which are exercises that you can work through to learn key concepts. And yes, they are programs you can write and execute in 10 minutes (unless you are a really slow typist, like me. In that case, they become “20 minute programs”.)

Academic Misconduct

In this class, the only way to truly learn the concepts to is do the work yourself. I encourage working with other people on the course concepts. When you begin to write the program, complete and submit your own work.

Work that has obviously been copied or in the more extreme case, when the original author’s name has not even been changed, both parties will receive a 0 grade for that assignment. Both parties will also be turned over to the Office of Judicial Affairs.
## Schedule

<table>
<thead>
<tr>
<th>#</th>
<th>Day</th>
<th>Date</th>
<th>Topic</th>
<th>Reading</th>
<th>Notes</th>
<th>Work Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>W</td>
<td>Sept 8</td>
<td>Introduction</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>M</td>
<td>Sept 13</td>
<td>Number Systems, Data Types, Variables, I/O</td>
<td>App. F, 2.1 - 2.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>Sept 15</td>
<td>Arithmetic and String Expressions</td>
<td>2.5 - 2.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>M</td>
<td>Sept 20</td>
<td>Objects</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>W</td>
<td>Sept 22</td>
<td>Flow of Control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>M</td>
<td>Sept 27</td>
<td>Functions</td>
<td>5.1 - 5.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>W</td>
<td>Sept 29</td>
<td>Review</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>F</td>
<td>Oct 1</td>
<td>Midterm Exam #1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>M</td>
<td>Oct 4</td>
<td>Software Design</td>
<td></td>
<td>5.7 - 5.13</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>W</td>
<td>Oct 6</td>
<td>Software Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>M</td>
<td>Oct 11</td>
<td>Classes</td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>W</td>
<td>Oct 13</td>
<td>Classes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>W</td>
<td>Oct 20</td>
<td>Adv. Flow of Control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>M</td>
<td>Oct 25</td>
<td>Testing and Debugging</td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>W</td>
<td>Oct 27</td>
<td>Review</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>F</td>
<td>Oct 29</td>
<td>Midterm Exam #2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>M</td>
<td>Nov 1</td>
<td>1Dim Vectors/Arrays</td>
<td>9.1 - 9.5.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>W</td>
<td>Nov 3</td>
<td>1Dim Vectors/Arrays</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>M</td>
<td>Nov 8</td>
<td>Multi-Dim Arrays</td>
<td></td>
<td>9.5.4</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>W</td>
<td>Nov 10</td>
<td>Review</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>W</td>
<td>Nov 17</td>
<td>Final Exam 1:00 - 3:00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Always have readings scheduled for that day complete prior to the class meeting

**Note:**
- M = Monday
- W = Wednesday
- F = Friday