CSC 6504  
Selected Topics: Software Engineering
Fall 2004

James Steven O’Neal
MCC 304
601-925-3994
jsoneal@mc.edu
www.mc.edu/~jsoneal

COURSE CREDIT: 3 hrs. credit  
PREREQUISITES: CSC graduate student

OFFICE HOURS: MWF 9:00-11:00; WR 2:00-3:00; MT 5:00-6:00; other times by appointment


The ACM Student Portal Package, for access to ACM’s Digital Library. Access to papers in ACM’s Digital Library is required. For access you must join ACM as a student member, see www.acm.org.

Papers as assigned by the instructor.

Optional: IEEE Computer Society’s Digital Library

DESCRIPTION: An advanced study of Software Engineering topics and literature

EVALUATION: The instructor reserves the right to make adjustments as necessary.

Course participation: Student participation in class discussion is worth 100 points. Students are expected to read assigned papers before class time and contribute to the material presented and discussed.

Research summaries: There will be research summaries required on selected assigned papers worth a total of 50 points.

Presentations: Each student will present a paper(s) to the class on an instructor approved topic. Presentations will be worth a total of 100 points.

Exams: There will be two take home exams, a midterm and a final, each worth 100 points.

Grading Scale:

<table>
<thead>
<tr>
<th>Score Range</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>450 - 500</td>
<td>A</td>
</tr>
<tr>
<td>400 - 449</td>
<td>B</td>
</tr>
<tr>
<td>350 - 399</td>
<td>C</td>
</tr>
<tr>
<td>300 - 349</td>
<td>D</td>
</tr>
<tr>
<td>0 - 299</td>
<td>F</td>
</tr>
</tbody>
</table>

Note: The student is expected to keep copies of all assignments. All graded work that is not returned to the instructor, should be retained by the student. If there were to be any question later about your grade, you would be expected to show these papers or provide a copy of the assignment.
CLASS ATTENDANCE:
The student is expected to attend classes. Regulations for class attendance are given in the Class Schedule. Remember in a night class, 3 absences is an automatic F. **Three tardies counts as one absence in this class.** (See the Mississippi College catalog).

MAKE-UP WORK & TESTS:
Students are expected to take tests on the day they are assigned. However, it is the student's responsibility to contact the instructor in case of an emergency illness or death in the family before the test. At that time the student and instructor will agree on a time for the make-up exam. This time should be within 2 days of the missed test. Assignments are to be turned in on the day they are due!! All work is due at the beginning of the class period. Any work not turned in will lose 10% credit for each school day until the third day. The due date at the beginning of class is day 1. No work will be accepted after the third day. Under no circumstances will work be accepted after the assignment has been graded and handed back in class. Laboratory assignments not completed during the period are due before the next laboratory meeting. A missed laboratory can be made up at a special laboratory session at the end of the semester.

ACADEMIC INTEGRITY:
This statement on academic honesty in computer science courses is an addendum to the Mississippi College policy 2.19 found at [http://www.mc.edu/publications/policies/academic/219.wpd](http://www.mc.edu/publications/policies/academic/219.wpd). In a computer science class individual effort is expected. Student misconduct not only includes cheating on tests, but also extends to copying or collaborating on programming assignments, projects, lab work or research unless otherwise specified by the instructor. Using other people's accounts to do your work or having others do your work is prohibited. Close proximity in lab does not mean collaboration is permitted. **NOTE:** Discussing logical solutions to problems is acceptable, exchange of code, pseudocode, designs, or procuring solutions from the Web, other texts, the Internet or other resources on or off campus is not acceptable. **First offense:** grade of 0 for **all** parties involved unless the "guilty" party can be determined as well as any punishment deemed necessary under policy 2.19 **Second offense:** grade of F in the course as well as any punishment deemed necessary under policy 2.19

SPECIAL ACCOMMODATIONS:
If you need special accommodations due to learning, physical, psychological, or other disabilities, please contact Dr. Buddy Wagner in the Counseling and Career Development Center. He may be reached by phone at (601) 925-3354 or by mail at P. O. Box 4016, Clinton, MS 39058.

DROPPING A COURSE: LAST DROP DATE - November 1
Students cannot withdraw after this date with a W (passing) unless the three following criteria are met:
- Extenuating circumstances (clearly outside the student’s control)
- Passing the course at the time of withdrawal
- Does not have excessive absences at the time of withdrawal

**NOTE:** Dropping after the THIRD (3rd) WEEK will result in a grade of W appearing on your permanent record (transcripts). See [http://www.mc.edu/publications/policies/academic/213.wpd](http://www.mc.edu/publications/policies/academic/213.wpd).
CSC 6504
Selected Topics: Software Engineering

6:00pm-8:45pm M
MCC 104
Spring 2004

Exam Dates

Midterm Exam .............................................................. October 18
Final Exam, 6:00pm .............................................................. December 13

The first part of this course will be driven by discussion of applying the SWEBOK and readings mostly from the tentative topics list. Papers will be assigned at least one week before class discussions. Your research summaries are intended to prepare you for this discussion. The topics for latter part of this course will be chosen from input from the class and the topics chosen for your presentations.

Tentative Topics

Model Driven Architecture
Object Constraint Language (of the UML)
Requirements Engineering
Reverse Engineering
Software Metrics
Software Maintenance
Software Project Catastrophes
Software Engineering Processes and Management
Software Quality
Team Techniques for Software Development