**SELF-ASSESSMENT OF USU CS COURSE**
**BY INSTRUCTOR**

**Course:** CS 6890  
**Semester:** Fall 2006  
**Instructor:** Jerry James

<table>
<thead>
<tr>
<th>List Course OUTCOME</th>
<th>Assessment tool and passing criteria</th>
<th>% Passing</th>
</tr>
</thead>
</table>
| Write simple multithreaded programs in either C with PThreads or in Java | Semester project, Homeworks  
Criterion: cumulative 70% or higher on relevant parts | 100% |
| Correctly describe and use key thread concepts, such as mutexes and condition variables | Semester project, Midterm 1, Final.  
Criterion: cumulative 70% or higher on relevant parts | 100% |
| Be familiar with at least one debugging technique for multithreaded programs | Semester project, Homework 3.  
Criterion: cumulative 70% or higher on relevant parts | 100% |
| Be familiar with at least one thread-related design pattern | Midterm 2, Final.  
Criterion: cumulative 70% or higher on relevant parts | 93.3% |
| Be familiar with multiple performance bottlenecks in multithreaded software and describe how to avoid them | Midterm 2, Final.  
Criterion: cumulative 70% or higher on relevant parts | 93.3% |

**Self assessments of strengths and weaknesses this semester (what worked, what did not work):**

I attempted to give the class heavy homework assignments at the beginning of the semester, then slack off the homework load as their semester projects got into full swing. However, this did not have the intended effect. The homework is very useful for test preparation purposes; all I accomplished was to make the class less prepared for the final than they had been for the first midterm.

I handed out lots of sample code to illustrate the various issues we discussed. Many members of the class told me that the code was very useful. I should probably print it up and bind it since the number of pages total is close to 100.

**Recommendations of changes to implement next offering:**

Balance the homework throughout the semester. It is too useful as a test preparation aid to slack off on it, even when the students are scrambling to finish their semester projects.

Print and bind the sample code to be given to the class, along with a table of contents, and possibly even an index showing where thread-related functions are invoked.

Figure out some way to goad the students into starting their semester projects sooner.