Salem STATE COLLEGE
Educating you for life.℠

Computer Science Department
cs.salemstate.edu\dept

SYLLABUS

CSC 202J Software Design and Programming II (using Java J2SE 6) **Fall 2010**
Prerequisites: CSC 201J with a grade of C+ or higher

Instructor: Beifang Yi **Office**: MH 208D **Phone**: (978) 542-7426
email: byi@salemstate.edu **Hours**: TR(10-12:15), W (1:00-4:30) **Web Site**: http://cs.salemstate.edu/~b_yi/

Section	Time	Room	Final Exam
01	T & R 12:30-1:45pm	MH 206	**Wednesday 12/15, 11:00am-1:00pm MH 206**
L21	T & R 2:00-3:15pm	MH 210	

Catalog description:
This course extends the treatment of object-oriented methodologies, languages and tools begun in CSC201J. The emphasis is on the analysis of complex problems, particularly those involving multiple design alternatives, and the use of class libraries. Specific topics include inheritance, polymorphism, recursion, stream and file I/O, exceptions, and graphical interface programming. Style, documentation, solution robustness, and conformance with specifications are emphasized throughout. Three lecture hours and three hours of scheduled laboratory per week plus extensive programming work outside of class.
Prerequisite: CSC201J with a grade of C+ or higher.

Course Goals:
The purpose of this course is to enhance and extend students' understanding of tools and techniques for object-oriented software development. Upon completion of the course, a student should be able to do the following:

- CG1: analyze a problem statement for completeness and clarity;
- CG2: use the methodology of object-oriented design to develop class diagrams (data descriptions and methods) for a problem solution;
- CG3: convert this solution into source code in the designated high-level programming language in accordance with a well-defined set o standards;
- CG4: debug and test the program;
- CG5: provide clear documentation for the result.

Course Objectives:

- CO1: Students will gain a deeper understanding of object-oriented design methodology.
- CO2: Students will learn to recognize situations in which multiple design alternatives are possible.
- CO3: Students will learn to recognize and apply design patterns.
- CO4: Students will gain experience in judging the effectiveness and cost of a software design.
- CO5: Students will gain experience in choosing among competing design alternatives.
- CO6: Students will gain experience in the use of the UML modeling language.
- CO7: Students will extend their knowledge of an object-oriented programming language, including graphical user interfaces, event-driven programs, file-based input/output and the use of libraries.
- CO8: Students will produce full documentation for several completed projects, including formal class diagrams.

Course Topics:

A detailed topics list and a general course bibliography can be found on the Computer Science Department website at http://cs.salemstate.edu/dept/index.php?page=184. Select CSC 202J to access a PDF document.

Text:

(Required) Java How to Program: Early Objects Version, 8th ed., by Deitel & Deitel, Prentice-Hall, 2010 (ISBN-13: 978-0-13-605306-4).

Required Material:

(Required) Thumb (flash) drive, 2 GB minimum

Software:

(Required) J2SE 6.0 and NetBeans 6.8 (this is the only IDE that will be covered in class). Free copies of the software can be downloaded in the Department labs - instructions will be given in class. DO NOT DOWNLOAD FROM www.netbeans.org - the version of NetBeans to be used has been heavily customized for use in this class. Older versions of NetBeans (6.7.1 and earlier) will generally **NOT** be acceptable.

Cell phones:

Turn the ringer off, or, better yet, *turn the phone off*.

Lecture Attendance:

Class policy is that of the Registrar's office - see the College catalog for details. Lecture will start promptly at the scheduled time, so please make a serious effort to not be late; if you *have* to be late, please be discrete when entering the classroom. While class attendance does not *directly* affect your final grade, some of the material covered in class is not found (in the same form) in the text, so class attendance and notes are very important. Note that you are at all times responsible for materials and assignments discussed in class: if you miss a class, try to get lecture notes from a classmate and review them **before** the next lecture, and **check your email or visit course homepage (**http://cs.salemstate.edu/~b_yi/2010Fall/CSC202J/index.html**) for any materials that may have been distributed**. Each student is responsible for completing all course requirements and for keeping up with all that goes on in the course (whether or not the student is present).

Quizzes will be held in class lecture hours.

Scheduled Lab Attendance:

Attendance during lab time is ***STRONGLY*** **recommended**. Lab will be used to review or present software tools, to discuss and investigate Java implementation details that time may not permit to be fully explored during the scheduled lecture period, for design and implementation drills, for occasional short lab (programming) exercises, to assist with design and debugging problems that arise in longer lab / project exercises, and to check/examine/grade the exercises and homework.

Student-Teacher Communication:

Learning how to develop software is very much a **hands-on, experiential process** - the only way to be sure that you understand the material is to apply it by designing and writing programs. The nature of programming is such that it is relatively easy to "get stuck" on minor technical topics that can be difficult to recognize, particularly at early stages of this course - this can lead to a significant amount of what feels like wasted time. While the single most effective way to deal with these problems is to talk to the course instructor, that approach can be problematic if the class meets only once or twice a week and/or if the instructor's office hours conflict with students' obligations.

If you have any questions regarding course material, and *in particular if you are having problems with a programming project*, the most effective way to get assistance is to **discuss with the instructor (either in the class or outside the classroom).** If you want to send the instructor an email about the problem, use the Subject line to succinctly state the problem/question, use the body of the email to describe in as much detail as possible what you need assistance with, and (if the problem relates to a Java project) attach a zip archive containing the *entire project* to the email. Note that attaching only the Java source file, or copying a chunk of your Java code and pasting it into your email, is basically a waste of time and communication packets. The *vast* majority of technical problems require that an *entire project* be provided to the person providing assistance - most technical problems can have many possible causes, and not having access to the complete project will mean that anyone providing assistance will have to do some guessing. Instructions on how to create a zip

archive will be presented and discussed in class - it is *very important* that you become familiar with this simple technique.

Final Grade:

Final grade will be determined using the following grading weights and formula:

	FS1	FS2
assignments (labs/project or short-answer exercises)	35%	55%
quizzes	10%	10%
midterm examination	22%	14%
final examination	33%	21%
Your final grade = Max (FS1, FS2)		

You will receive two final scores (i.e., FS1 and FS2 in the above table) based on your course work (labs/projects, quizzes, midterm and final examinations). Your final grade will be the higher score point of these two final scores.

- The distributed weights for FS1 are: assignments (labs/project or short-answer exercises) will account for 35%, quizzes for 10%, midterm examination for 22%, and final examination for 33%.
- The distributed weights for FS2 are: assignments (labs/project or short-answer exercises) will account fo55% , quizzes for 10%, midterm examination for 14%, and final examination for 21%.

Attendance is not used to calculate the final grade: however, note that you are at all times responsible for assignments and materials presented in class.

The following table shows how the course work is assessed against the course objectives:

	Quizzes	Assignments (including programming projects and lab exercises)	Exams
CO01	✓	✓	✓
CO02	✓	✓	✓
CO03	✓	✓	✓
CO04	✓	✓	✓
CO05	✓	✓	✓
CO06	✓	✓	✓
CO07	✓	✓	✓
CO08		✓	

Assignments (Laboratory / Project/Short-Answer Exercises):

10-15 exercises will be assigned during the semester. *Most will have pre-lab activities to be completed prior to the implementation of the assigned tasks while a few will be in the form of short answer*. Exercises *will definitely require **significant** programming time outside of scheduled lab*. Submission requirements and mechanics will be stated on each exercise In general, each exercise will have an assigned due date and time: the required material(s) are to be submitted no later than midnight of that date. Please refer to Final Grade above for the grading weight of the assignments.

One assignment with the lowest grade will be dropped from the final grading/scoring.

Exams/Quizzes:

There will be two exams, a midterm (usually in week 8) examination and a *comprehensive* final examination. There will be about 6 to 10 quizzes to be held in class lecture hours (one quiz with the lowest grade will be dropped from the final grading/scoring).

Please refer to Final Grade above for the grading weights of the exams and quizzes.

Missed Tests:

Tests (exams and quizzes) may not be made up except for *documented emergency* situations. If a test must be made up, arrangements must be made with the instructor to take the test before it is discussed in class (usually within a week of the test being administered).

Homework:

Readings will be assigned from the text on a regular basis: for the maximum benefit from reading, do the readings before the material is covered in class. Supplementary material will be distributed on a regular basis, *and will be the primary focus of class discussions.* In particular, at least one complete, functioning example Java application project will be distributed per week or biweekly: these projects include extensive student-oriented documentation/comments that are designed to guide students through the process of learning how to design and implement software. Occasional worksheets and problems will be assigned and collected as a portion of quizzes.

Due Dates:

- **There will be a 5% penalty for each day (including weekends, holidays) an assignment (lab/project/short-answer exercise) is late**; penalties accrue at 12:00 midnight of the assigned due date.
- All homework must be in by the beginning of lecture on the assigned due date - late submissions will **not** be graded.

Study Groups:

While I strongly encourage study groups, for non-group assignments I require that each student hand in his/her answers in his/her own words **- if two answers come out exactly the same or highly similar, neither will receive credit and/or further actions will be taken** (such as reporting to the department and/or college). Given the nature of most of the projects, homework questions and writing assignments, it will be almost impossible for two people to come up with highly similar answers UNLESS they copy.

Academic Integrity:

Academic Integrity Policy and Regulations can be found in the College Catalog and on the College's website (http://www.salemstate.edu/content_images/academic_integrity_regulations_2007(1).pdf). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the College's Policy* and will be dealt with according to the College's formal Procedures.

"Salem State College is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately. Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services."

In the event of a college declared critical emergency, Salem State College reserves the right to alter this course plan. Students should refer to http://www.salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a college declared critical emergency. In the event of an emergency, please refer to the alternative educational plans for this course located at http://cs.salemstate.edu/~b_yi/2010Fall/CSC202J/emergency/index.html. Students should review the plans and gather all required materials before an emergency is declared.

Please remember that if, for any reason, you decide to drop this course, you **MUST** do so officially through the Registrar's office. The last day to withdraw from a course this semester is **November 19th**.

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via all regular class communication mechanisms (class discussion, emails, and/or the instructor's website at http://cs.salemstate.edu/~b_yi/.).