Salem STATE COLLEGE
Educating you for life.℠

Computer Science Department
cs.salemstate.edu\dept

Syllabus **Spring 2010**

CSC 201J Software Design and Programming I (using Java J2SE 6) **4.0 cr. [DII]**
Prerequisites: high school algebra I & II, experience with a window-based operating system and the use of email and a word processor.

Instructor: Beifang Yi **Office**: MH 208D **Phone**: (978) 542-7426
email: byi@salemstate.edu **Hours**: TR(11-12:30), WF (11-1:30) **Web Site**: http://cs.salemstate.edu/~b_yi/

Section	Time	Room	Final Exam
01	T/R 12:30-1:45pm	MH 206	**Friday 5/7, 11:00am-1:00pm MH 206**
L21	T/R 2:00-3:15pm	MH 210	

Section	Time	Room	Final Exam
02	W/F 1:30-2:45pm	MH 206	**Tuesday 5/11, 2:00pm-4:00pm MH 206**
L22	W/F 3:00-4:15pm	MH 210	

Catalog description:

This course introduces a set of fundamental design principles and problem-solving techniques for the development of computer algorithms and their implementation as programs. Problem solutions are developed with the help of an appropriate modeling language and then coded in an object-oriented programming language. (Consult the Computer Science Department for the languages and tools currently in use.) Topics such as problem specification, object-oriented analysis and design, standard data types, control structures, methods and parameter passing, and design for reuse are presented through a study of specific example problems and solutions. 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.

Prerequisites: High school algebra I & II; experience with a window-based operating system and the use of email and a word processor.

Course Goals:

The purpose of this course is to develop students' understanding of a coherent set of tools and techniques for creating computer solutions to simple problems in data manipulation. 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 of style rules;
- CG4: debug and test the program;
- CG5: provide clear documentation for the result.

Course Objectives:

Upon successful completion of the course, a student will have:

- CO1: demonstrated knowledge of the syntax elements of an object-oriented programming language;
- CO2: gained experience in analyzing problem statements for completeness and consistency;
- CO3: practiced standard techniques of problem analysis;
- CO4: applied the fundamentals of object-oriented design methodology;
- CO5: learned and utilized simple techniques for validation and verification of programs;
- CO6: created full documentation for several completed projects.

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 201J 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 that have been customized for the course can be downloaded in the Department labs - instructions will be given in class.

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, much 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 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).

Scheduled Lab Attendance:

Attendance during lab time is **required**, *not optional*. 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.

Lab attendance constitutes **5%** of the final grade. Each student should attend at least **20 lab meetings** to obtain this lab attendance grade. **Missed lab meetings can only be made up through individual projects** (different from the exercises and homework) unless *under extreme conditions/emergency with the proper documentation*.

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:

assignments (labs/project or short-answer exercises)	50%
lab attendance	5%
midterm examination (written test & midterm project)	20%
final examination (written test & final project)	25%

Laboratory / Project Exercises will account for 50% of the final grade and lab attendance for 5%. There will be a midterm exam accounting for 20% and a final exam accounting for 25% with **each having two portions**: one written test (closed-book, closed-notes, no-electronic-devices, in the lecture hours or final exam time) and one semester project (in the lab hours).

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

12-15 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*. Initial exercises are designed to be completed during a single lab session; as the semester progresses, exercises will may require more than a single lab session *and will definitely require programming time outside of scheduled lab*. Submission requirements and mechanics will be stated on each exercise. Each exercise will have an assigned due time: any required material(s) are ***to be submitted via email or in hardcopy/printout or to be checked in the lab* depending on the nature of the exercises (please read the submission instruction carefully).** These exercises are worth a total of 50% of the final grade.

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

Exams:

There will be two exams, a midterm (usually in week 8) that is worth 20% of the final grade and a *comprehensive* final exam that is worth 25% of the final grade, with each having two portions: one written test (closed-book, closed-notes, no-electronic-devices, in the lecture hours or final exam time) and one semester project (in the lab hours).

Missed Tests:

Tests 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, and graded on a check/no check basis: this homework must be turned in by the beginning of the lab on the assigned due date. **Late homework will NOT be accepted**. The grade for this part will be used as **a bonus added to the final grade** listed above. The maximum grade of this part will be **2 ~ 5 percentage points** (the exact number of points will be announced around the midterm week).

Due Dates:

- **There will be a 6% penalty for each day (including weekends, holidays) an assignment (lab/project/short-answer exercise) is late**.
- All homework must be in by the beginning of the lab on the assigned 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."

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 **April 16th**.

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/.).