Computer Science Department
cs.salemstate.edu

SYLLABUS **Summer II, 2016**

CSC 202J Software Design and Programming II **4 credits, DII**
Prerequisites: CSC110 or CSC 201J

Instructor: Beifang Yi **Office**: MH 211A **Phone**: (978) 542-7246
email: byi@salemstate.edu **Hours**: 3:50-4:20pm (TR) **Web Site**: http://cs.salemstate.edu/~byi/
10:20pm-10:50pm (TR)

Section	Time	Room	Final Exam
22	T & R 4:20pm—10:20pm	MH 206 and MH202	**Thursday 8/18, 8:20pm-10:20pm MH 206**

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:

- CO01: gained a deeper understanding of object-oriented design methodology;
- CO02: learned to recognize situations in which multiple design alternatives are possible;
- CO03: learned to recognize and apply design patterns;
- CO04: learned and utilized techniques for validation and verification of programs;
- CO05: gained experience in judging the effectiveness and cost of a software design;
- CO06: gained experience in choosing among competing design alternatives;
- CO07: l gained experience in the use of the UML modeling language;
- CO08: extended their knowledge of an object-oriented programming language, including graphical user interfaces, event-driven programs, file-based input/output, and the use of libraries;
- CO09: produced full documentation for multiple completed projects, including formal class diagrams;
- CO10: participated in one or more group 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 115 to access a PDF document.

Text:

(Required) Java How to Program: Early Objects, 10th Edition, by Deitel & Deitel. Prentice-Hall, 2015 (ISBN: 978-0-13-380780-6).

Required Material:

(Required) Thumb (flash) drive, 4 GB minimum or online storage.

Software:

(Required) J2SE 7.0 (or above) and NetBeans 8.0 (or above) (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/Lab Attendance:

Class policy is that of the Registrar's office- see the University catalog for details. 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. We will use SSU's online course management system, Canvas (https://elearning.salemstate.edu/) to post assignments, quiz grades, and announcements regarding the course topics and progress. You will need to visit Canvas (with your SSU Navigator use-name and password) for the course activities. Canvas uses your ***SSU-stored email*** for the communication between you and the instructor and thus you ***must use this email address***. 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).

Class attendance is ***strongly recommended***. Some of the class hours will be used to review or present software tools, to discuss and investigate Java implementation details that time may not permit to be fully presented in the texts (for design and implementation drills, for programming exercises,) to assist with design and debugging problems that arise in longer lab / project exercises, to have some of the course quizzes (that is, programming-related quizzes), and to check/examine/grade the exercises and homework.

Student-Instructor 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).*

Final Grade:

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

	Grade-A	Grade-B
assignments (programming project or short-answer exercises)	45%	45%
quizzes	25%	0%
final examination	30%	55%
final grade (overall)	Max(*Grade-A*, *Grade-B*)	

Two different grading formulae are used to calculate your semester overall final grade: *Grade-A* and *Grade-B* and your final grade will be the larger of them. Note: it is ***more difficult*** to get same grade through ***Grade-B*** than through Grade-A; it is easier to get higher grade through Grade-A scheme.

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	✓	✓	✓
CO09		✓	
CO10		✓	

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

About 7-10 exercises (including short-answer exercises, programming projects) 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.

There is a deadline to each assignment and penalty will be imposed for late submissions (see the Due Dates/Time below).

All the assignments must be submitted at Canvas (https://elearning.salemstate.edu/) unless otherwise noted

*Each assignment may have different full score points, depending on the difficulty and the amount of the work of the exercises. There will be one given in the semester and these extra assignments will be used as make-up assignments. Any assignment may be used as the extras/make-ups. The average score for the overall assignments will be the total scores received for all the assignments divided by the total scores of the required assignments. For example, suppose that there will be about 1100-point assignment questions given in the whole semester and that the required assignment total scores will be about 1000 points. You may not move the extra points to the final grade. For example, students A and B have completed 1050-point and 800-point assignments and their Semester Assignment Grades will be 1000/1000 * 45 = 45 and 800/1000* 45 = 36 points respectively (suppose that the required assignment points is 1000).*

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.

Exams/Quizzes:

There will one *comprehensive* final examination.

There will be about 4~6 quizzes to be held in class hours (one quiz with the lowest grade will not be used to calculate the final grading/scoring). There are different forms of quizzes: *short-answer questions* (paper-based or coding practice questions) and complete *Java Programming* quizzes.

The **Java Programming Quiz**: *This form of quiz is in the format of Java coding project and will be held in the lab and completed on the lab machines.*

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).

Due Dates/Time:

- **There will be a 30% penalty for each week an assignment (lab/project/short-answer exercise) is late**; penalties accrue at the due time of the assigned due date (any assignments being submitted later will be considered "being late at least one week", that is, one week late is defined as Day 1 after submission deadline, up to and to including Day 7; ***n***-weeks-late can be counted in the same way.).
- **No assignments will be received after the final examination**.

Study Groups:

While I strongly encourage study groups, I require that each student hand in his/her answers in her/his own words - if two answers are highly similar to each other, neither will receive credit.

When working on your programming projects, you may discuss with others the project topics, the algorithms and methodologies related to the project; but when you work on writing the code, this coding work should be 100% of your own work. **If two answers/written code segments 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 university). 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 University Catalog and on the University's website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). 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 University's Policy* and will be dealt with according to the University's formal Procedures.

All students are expected to be familiar with the academic regulations, including those regarding Academic Integrity, for Salem State University as published in the college catalog. In addition, 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).

Salem State University 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 university declared critical emergency, Salem State University 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 university declared critical emergency. In the event of an emergency, please refer to related course announcement and alternative study guide and materials at Canvas (https://elearning.salemstate.edu/) by logging to the course link at Canvas. 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 **August 1st**.

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 course link at Canvas https://elearning.salemstate.edu/).