

# CS 225 Computer Science II with Lab

## Syllabus, Fall 2017

### Embry Riddle Aeronautical University, Daytona Beach Campus

#### Course Information:

Course – Section	CS 225 - 01	CS 225 - 02	CS 225L - 01	CS 225L - 02	CS 225L-03
<b>Meets</b>	MWF 9:00 – 9:50 IC 104	MWF 10:00 – 10:50 COB 228	Tu 12:00 – 2:00 LB 264	We 1:00 – 3:00 LB 164	Th 2:15 – 4:15 LB 164
<b>Text</b>	Liang, Y.D. <i>Introduction to Java Programming</i> Note: If you will take SE 320 at some point, buy the full, 33 chapter version, otherwise buy the brief version.				
<b>Supplemental Materials</b>	Online lab manual, posted on Canvas. Any Java text and online resources (Oracle help pages, Stackoverflow forums, etc...)				
<b>Tutoring</b>	TBD				
<b>Instructor(s)</b>	Keith Garfield garfielk@erau.edu 386.226.7081 Office LB 356	Keith Garfield garfielk@erau.edu 386.226.7081 Office LB 356			
<b>Office Hours</b>	MWF 11:00 -1:00	MWF 11:00 -1:00			
<b>Pre-Requisite</b>	CS 223 or EGR 115				

#### Course Catalog Description:

This course emphasizes program design, style, data abstraction, information hiding, and testing; advanced programming features; and introduction to object-oriented concepts, basics of algorithm analysis, exception handling, string processing, recursion, pointers, and simple data structures. The course has a closed laboratory that includes activities dealing with the computing environment, the software development process, and programming exercises.

#### Course Overview and Objectives:

This course is an introduction to object-oriented programming and software engineering. The purpose of this course is to familiarize students with mid-level principles of software design and construction, within an object oriented programming environment.

It is assumed that students have some experience with a current high-level language, but no prior knowledge of object-oriented principles is required. The coursework will emphasize “hands-on” application of course material through in class exercises, programming assignments, laboratory sessions, and a software project.

Students who master the course material will be able to:

1. Read and write algorithms in pseudo-code.
2. Develop basic software requirements with appropriate test conditions.
3. Perform basic software testing and debugging.
4. Document software design, test sequences, and source code.
5. Create UML class diagrams describing software designs.
6. Design software classes based on relatively simple specifications.
7. Design and implement object-oriented applications containing inheritance, abstraction, and interfaces.
8. Design for and implement software reuse.
9. Create simple graphical user interfaces.
10. Create and manipulate text files.
11. Implement exception handling techniques.
12. State principles of recursion and design recursive algorithms.

**Labs:**

This course includes up to 11 two-hour laboratory sessions. The laboratory exercises will explore topics related to software engineering that enhance or extend the material covered in lectures. Lab times at the end of the semester are reserved for assisting students in completing their course project.

**Grading:**

There are 1000 points available in this course, distributed per the table below. Overall course grades are assigned as A (900-1000 points), B (800-899 points), C (700-799 points), D (600-699 points) or F (below 600 points).

Note: To pass this course the course project must be completed. Failure to do so will result in a failing grade.

Component	Points	Notes
Homework	350	Late NOT accepted.
Labs	350	Per lab schedule.
Project	300	See Project Assignment.
Extra Credit	50	Up to 5% of total points, some in class.
Total	1000	

**Class Policies:**

ACADEMIC INTEGRITY: It is not acceptable or allowable to develop significant portions of programs as a group effort, or to copy software from online sources, or to submit assignments that have not been created independently by the student.

Honesty and integrity are expected of all students and faculty at ERAU. Dishonest conduct will result in sanctions up to receiving a failing final course grade. Refer to your Student Handbook and Department definition of academic integrity for additional guidance. You are encouraged to discuss homework material and methods with others, however, the homework submitted must be your own original work. Note: changing the variable names does not constitute original work.

LECTURES: Attendance is strongly encouraged, but is not strictly required, nor calculated directly as part of your grade. Lectures will follow general topics in the reading assignments, but may also introduce new material. Lectures will not necessarily cover all aspects of the reading assignments. Students are expected to read all material covered in the course.

HOMEWORK: Homework assignments are provided and submitted electronically via Canvas. No late homework will submissions will be graded. All homework assignments must be properly formatted as stated in the assignment. The homework may introduce material not presented in reading assignments or lectures. Students may be excused from homework deadlines or examinations only through prior coordination with the instructor, or through instructor approval in the event of a medical or personal emergency sanctioned by the University Office of Judicial Affairs.

COMPUTERS/CELL PHONES: These items are NOT to be used in class without instructor permission.