

SENG 6245: Software Construction

Spring 2019

Instructor	Mark Hills
Scheduled Class Time	Class: Tuesday and Thursday, 12:30pm to 1:45pm Class meets in Brewster B-204 (Global Classroom)
Instructor Office	Science & Technology Building, C-110
Office Hours	Tuesday 10:00am – 11:30am Wednesday 1:00pm – 3:00pm Thursday 10:00am – 11:30am Or by appointment
Instructor Phone	252-328-9692
Instructor Email	hillsma@ecu.edu , responses within 24 hours during the week, potentially longer during holidays or weekends
Course Web Page	https://blackboard.ecu.edu/
Required Textbooks	None (see below)

Course Description and Objectives

The catalog description for this course is as follows:

Application of software specifications, design patterns, object-oriented design and concurrent programming, and testing techniques for designing, constructing, and testing large-scale software systems.

The course teaches software construction concepts including object-oriented programming, configuration management, and the use of automated unit testing and program analysis tools. The course discusses and illustrates concepts such as modularization, specification, information hiding, concurrency, abstraction, generics, design patterns, and unit testing. The purpose of this course is to give the students a solid understanding of modern software construction, and to prepare students to construct high quality programs.

After taking this course, you should be able to:

- develop software using sound software development principles, tools, and techniques;
- create object-oriented systems that make effective use of abstraction and specification concepts, including the use of appropriate interfaces, classes, and methods, and the effective use of concepts such as concurrency, generics, inheritance, and polymorphism;
- understand basic concepts of automated software testing, and perform automated software testing using unit testing and continuous integration tools;
- make use of continuous integration, containerization, cloud, and automated deployment solutions for automating the integration and release of software systems.

Although these concepts are covered mainly in the context of Java, they apply to most modern programming languages and software systems.

Prerequisites

The prerequisite for this course is SENG 6230 or the consent of the instructor. You are expected to be familiar with Java at the level of a student that has successfully completed the material typically covered in an introductory computer science sequence, such as the material in SENG 5000. If you are familiar with a similar language, especially C#, adapting to Java should be fairly straightforward. If you are not familiar with Java, please read, and go through the exercises in, Schildt's "Java: A Beginner's Guide". You should also look at the posted Lynda videos, especially "Java 8 Essential Training". If you do not have any programming experience, you must take SENG 5000 before taking this course, this is **not** an introductory programming course. If you ignore this prerequisite, you should not expect to do well in this class.

Textbooks

There are four textbooks for the course, one recommended and three optional.

The recommended textbook is *Program Development in Java: Abstraction, Specification, and Object-Oriented Design*, by Barbara Ryder and John Guttag. This book is available through Amazon.com at <https://www.amazon.com/Program-Development-Java-Specification-Object-Oriented/dp/0201657686/> and should also be available through the ECU campus bookstore. You can also access this book using the Safari online book service, which is available for free if you are an ACM member. This book covers an older version of Java, so we will discuss how newer features in the language impact the examples in the textbook.

The optional textbooks are

- *Code Complete: A Practical Handbook of Software Construction*, 2nd edition, by Steve McConnell. This book is available through Amazon.com at <https://www.amazon.com/Code-Complete-Practical-Handbook-Construction/dp/0735619670> and should also be available through the ECU campus bookstore. You can also access this book using the Safari online book service, which is available for free if you are an ACM member. I will occasionally mention specific chapters or topics from this book.
- *Java: A Beginner's Guide*, 7th edition, by Herbert Schildt. This book is available through Amazon.com at <https://www.amazon.com/Java-Beginners-Seventh-Herbert-Schildt/dp/1259589315/> and should also be available through the ECU campus bookstore. You can also access this book using the Safari online book service, which is available for free if you are an ACM member. I will occasionally reference a chapter from this book as material that would be useful for you to review if your knowledge of Java in that area could use some reinforcement.
- *Effective Java*, 3rd edition, by Joshua Bloch. This book is available through Amazon.com at <https://www.amazon.com/Effective-Java-3rd-Joshua-Bloch/dp/0134685997/> and should also be available through the ECU campus bookstore. You can also access this book using the Safari online book service, which is available for free if you are an ACM member. We will be discussing specific items from this book in class.

Other helpful material, including references to books, conference or journal articles, tutorials on the web, and videos will be posted as the course progresses. You should read/view these when they are assigned so we can discuss them in class.

Exams

The final exam for the course will be on **Tuesday, April 30th**, from **11am – 1:30pm** in our normal classroom. More details about the exam will be available closer to the exam date. The exam will be a closed book exam, but you will be allowed to bring one page (letter size, front and back) of hand-written notes. We will review for the exam on Thursday, April 18th, which is the last day of class.

There will not be a midterm exam. Instead, we will have a number of hands-on activities over software construction topics, outside of the normal homework assignments, throughout the semester.

If you are taking the course online, you must have a proctor for the final exam. You must use the University of North Carolina Proctoring Network. More information can be found at: <http://online.northcarolina.edu/exams/overview.htm>

Grading

Students will be evaluated based on a combination of homework assignments, completion of the hands-on activities, the research project/lesson, and the final exam. The following grade cut-offs, using a 100-point scale, will be used:

Grading	
A	≥ 90
B	≥ 80
C	≥ 70
F	< 70

This grade is based on the following relative weights of the various activities:

Weighting	
Homework	40%
Hands-On Activities	20%
Research Project/Lesson	20%
Final Exam	20%

Homework assignments will be due roughly every two weeks. Hands-on activities will be issued periodically throughout the semester. More details about the homework, the hands-on activities, and the research project/lesson will be made available during the course.

Attendance Policy

On-campus students are expected to attend class (international students studying under a student visa are required to attend class). Online students are expected to keep up with the lecture videos to ask timely questions and participate in group chats/discussions of the material. You are responsible for announcements given in class or posted to Blackboard. If you are an on-campus student and you miss a class, you should watch the related video before the next class. Excuses that you did not

know about something because you did not come to class and did not see announcements on Blackboard will not be accepted. If you are having trouble understanding the lectures and/or assignments, come to office hours, schedule an appointment, or otherwise ask for help. Get help as early as possible. If you wait until the end of class to seek help, there is most likely very little that you can do to improve your score.

Starfish

This course may use the Starfish system to provide you with information on your performance within the course. For more information, please see <http://www.ecu.edu/cs-acad/advising/upload/Starfish-Student-Getting-Started.pdf>.

Student Conduct

Smoking is not permitted in classrooms. Please turn off mobile phones in class. Laptops and tablets can be used for taking notes, but they should not be used for other work (or recreational browsing, playing games, watching Netflix, or other activities not related to class).

Students are expected to abide by the university's Student Honor Code. The homework, including programming assignments, hands-on activities, and the research project/lesson, that you do is a critical part of your education. Each student is expected to do his or her own work, except where teamwork is explicitly allowed or required. That does not mean you are not allowed to discuss your ideas with other students. Working in groups can be beneficial, and I encourage you to talk through ideas with other students. But outright copying, either from other students or from other sources (e.g., textbooks, websites) is plagiarism and is unacceptable. Students who copy other students' work, or who allow their work to be copied, or who copy their work from other sources, such as the internet, are violating the ECU academic integrity policy. Not only that, if you are copying your answers instead of doing the work yourself, you are essentially missing the entire point of this course, which will come back to haunt you when you don't know this material at a future employer.

Other potential academic integrity violations are cheating, falsification, multiple submissions of the same work in different classes, and attempts at any of these violations. Please see http://www.ecu.edu/cs-studentlife/policyhub/academic_integrity.cfm for more details.

Academic integrity violations can result in a grade penalty up to and including an F for the course and a report of the incident to the university. Cheating on an assignment can result in negative credit for the assignment (instead of just a 0, you can lose the same number of points you could have gained legitimately doing the work).

Other Policies

No incompletes will be issued in this course except for extraordinary circumstances, and even then, only if you are nearly done already, and have done work of acceptable quality, so that you have a realistic change to pass the course.

All homework solutions, hands-on activities, and submissions for the research project/lesson, are due by the posted due date and time. Late submissions will not generally be accepted. If for some reason you are not able to complete the assignment on time, you must contact me directly with an explanation and request an extension before the deadline. If something comes up and you are

having trouble keeping up with the class, talk to me right away, *don't wait until the end of the semester!*

Course participation is an important part of the course. If you do not participate, you will make it harder to have the kinds of discussions we need to make the class interesting. Please read any assigned readings or watch assigned videos in a timely fashion, do the homework promptly when it is made available (so you know if you are going to get stuck!), and come to class prepared to talk.

Success in the class is directly correlated with class attendance, so I highly recommend that you attend and actively participate. If for some reason you cannot attend, please let me know – my expectation is that you will watch the lecture online and ask me questions about the material if you have any. For online students, I recommend that you watch the lecture the day it is given and send any questions before the next class session (so I can address them in class). Falling behind will make the course more difficult than it would otherwise be. I will be taking attendance at regular points in the class for my own records.

All code, test scripts, and other software artifacts must be stored in GitHub. I will not accept programming assignments submitted through Blackboard or emailed to me. If you have questions about your code, check it in to the related GitHub repository, that way I can easily look at it. Sending screenshots of your code is generally not helpful.

Course materials, including programming assignments and lecture notes, can only be publicly shared or used for commercial purposes if given permission. This is covered by ECU copyright regulations, available at <http://www.ecu.edu/pr/10/40/02>, which state the following:

7.1.3. Notes of classroom and laboratory lectures, syllabi, exercises and other course materials taken by Students shall not be deemed Student Works, may only be used for personal educational purposes, and shall not be used for commercialization by the Student generating such notes or by any third party without the express written permission of the author of such Works. Violation of University Policy may be grounds for disciplinary action pursuant with the ECU Student Conduct Process.

Weather Emergencies

In the event of a weather emergency, information about ECU can be obtained through the following sources:

ECU emergency notices	http://www.ecu.edu/alert
ECU emergency information hotline	252-328-0062

Students with Disabilities

East Carolina University seeks to comply fully with the Americans with Disabilities Act (ADA). Students requesting accommodations based on a disability must be registered with the Department for Disability Support Services located in Slay 138 ((252) 737-1016 (Voice/TTY)).

For more information, please see <http://www.ecu.edu/cs-studentlife/dss/>.

Caveats

Occasionally, it may be necessary to revise this syllabus due to extenuating circumstances. I reserve the right to revise this syllabus if the need arises. If I do so, I will announce the changes on Blackboard and/or in class.