Java Programming For Engineers and Scientists II, Fall 2021

David Anderson | danderson@tacomacc.edu
CS 143 Section OL1 (7913) | 5 Credits
September 20, 2021 – December 9, 2021
Canvas Class Direct Link: https://tacomacc.instructure.com/courses/2139015

Contents

Professor Information

Contact and Office Hours

Office hours will be conducted remotely on Zoom:

Days/Time: Monday – Thursday: 1pm-2pm, Friday: 9:30am-11am

Invite Links:

Monday – Thursday: https://tacomacc-edu.zoom.us/j/85945778146?pwd=b2JPR2RINWx1a1IvcGJydGJYQmdLUT09

Friday:
 https://tacomacc-edu.zoom.us/j/83673624317?pwd=MllmRTZYKzVBcTBMSWpnRHh6bWtUZz09

Password: 783774

You are encouraged to stop by Zoom office hours to ask questions about course material. I am always happy to provide help and additional explanation. If you are unable to ask your questions during office hours, the best way to contact me for help is email.

Communication Policy

Outside of Zoom office hours, my preferred method of communication with you is email, which I check frequently. You can email me at any time, but you may not receive a response outside regular business hours. Business days are Monday–Friday, 9am-5pm except for holidays. Generally, you can expect a response from me within 24 hours except in extreme circumstances.

Please set Canvas so that it sends you a message when I make a comment on your work. To do this, go to Canvas -> Account -> Notifications -> Course Activities -> Submission Comments. Click on the check mark for immediate notification to your home email or text, depending on what you prefer. You may also want to take a look at the other possible notifications and see what you would like to receive.

General Class Information

Class Information (Catalog Description)

CS 143 (Java Programming for Engineers and Scientists II) is an intermediate programming course, using Java. Topics will include classes, interfaces, inheritance, polymorphism, exception handling, recursion, data structures, and an introduction to performance analysis and implementation trade-offs.

Course Modality

This course is fully online and asynchronous. **Fully online** means that all course material will be available on Canvas and all assignments will be completed and submitted on Canvas. **Asynchronous** means that there will be no "live" lecture sessions. All lectures will be pre-recorded and posted on Canvas. **We will, however, have an on-campus proctored final exam.**

Face-to-Face Testing

There will be a face-to-face proctored final exam in this course. The exam will be in building 15, room 201 from 11:30am-1:30pm on Tuesday, December 7. If you are unable to attend the in-person exam, please let me know immediately so that we can find alternative arrangements.

Schedule
Generally, I will post 5 lectures each week (one for each business day of the week). You will have weekly exercises and assignments that test your knowledge of lecture topics. It is important to keep up with posted lectures, assignments, and exercises and not fall behind.

Prerequisites
CS 142 with a minimum grade of C or assessment above.

Textbooks & Supplemental Materials
All materials for this course are freely available online!

Readings: Assigned readings will come from a combination of "OpenDSA" by Karavirta, Shaffer et al (customized), and "Introduction to Computer Science using Java" by Bradley Kjell, available on Canvas. Additionally, we may reference other free websites and readings including "Java Structures" by Duane A. Bailey and "Introduction to Java Programming Version 7" by David J. Eck. Generally, readings are organized into modules on Canvas which will be linked to on the course home page.

Technology Required:

1. Canvas: we will be using this Learning Management System for sharing documents like this Syllabus, assignments, worksheets, etc. Your gradebook is there, too. You will be able to access Panopto-recorded lectures from Canvas. We will also use it for electronic submission of assignments.
2. You are going to need a modern computer for this class. Java is platform-independent – students have been able to use their own Linux or Mac system as well as the Windows systems that I'll use.
3. Java JDK (Java Development Kit): Please use JDK 8. I suggest using Amazon Corretto 8 which can be downloaded from here: https://docs.aws.amazon.com/corretto/latest/corretto-8-ug/downloads-list.html
4. Eclipse IDE for Java Developers is the IDE we will be using to write Java code in this course.

Learning Outcomes

Degree Learning Outcomes:
The abbreviation following each objective refers to the Degree Learning Outcomes: COK=Core of Knowledge; COM=Communication; CRT=Critical Thinking; IIT=Information and Information Technology; LWC=Living and Working Cooperatively; RES=Responsibility and Ethics

Engineering Program Learning Outcomes:
1. Apply knowledge of mathematics and science to engineering related problems. (COK, CRT)
2. Design a system, component, program or process to meet desired needs. (COK, COM, CRT, IIT, LWC, RES)
3. Conduct scientific experiments, analyze and interpret the resulting data. (COK, CRT)
4. Communicate design ideas, solutions to engineering related problems or results of scientific experiments effectively, using both English and mathematical languages. (COK, COM, CRT, IIT)
5. Function effectively on a team to produce a cohesive and professional work product. (COK, COM, CRT, IIT, LWC, RES)

Course Learning Outcomes:
Upon satisfactory completion of the course, students will be able to:

1. Utilize interfaces and inheritance appropriately in a moderate-size program. **PLO:** 1, 2
2. Implement stacks, queues, linked lists, and binary trees. **PLO:** 1, 2
3. Implement linear and binary search algorithms. **PLO:** 1, 2
4. Implement various sorting algorithms.
5. Analyze an algorithm using Big-O notation. **PLO:** 1, 4
6. Design, implement, document and test a medium sized program (about 1,000 lines) incorporating the concepts of modularity, encapsulation, and information hiding. **PLO:** 2, 3, 4
7. Write programs that implement user-interfaces reflecting current human factors expectations. **PLO:** 2, 4
8. Participate in a team to define, specify, implement, test and document a project of greater complexity than a single programmer could be expected to do within a specified time frame. **PLO:** 5

TCC Student Policies

TCC works hard to create a vibrant learning culture where you can succeed. Please refer to the college-wide standards and policies that support this important work. Some of these policies include Religious Accommodations, Withdraw/Retakes, and the Student Code of Conduct.

Class Policies and Procedures

Attendance

Attendance is not tracked in this class. All lectures are asynchronous (meaning you can view them when convenient for you). There are, however, regular deadlines for assignments and exercises that make keeping up with lectures essential.

Due Dates / Late work

All assignments are due by 11:59pm the day of the due date, unless otherwise listed on Canvas. Please see the **Graded Items** section on page 5 for the late work policy which differs depending on the type of assignment.

Classroom Concerns/Disputes/Final Grade Appeal Process

If you have questions or concerns about this class, please talk to me about them. If we are unable to resolve your concerns, you may talk next with the Chair, Ivan Ramirez Carrascal (iramirez@tacomacc.edu). He can assist with information about additional steps, if needed. If you think that your final grade has been given in error, please see the final grade appeal process.

Graded Items

Assessment Structure and Late Work

1. **Programming (50% of your grade):** I will post programming assignments on the class web site. The assignments will be posted in Canvas and **must be submitted using the Canvas assignment submission mechanism**. The assignment may be updated with announcements on Canvas. Important: see below under Academic Dishonesty about the importance of originality in your programs. You should be prepared to explain your assignments if I ask. **Note on deadlines**: Software is notorious for missed deadlines. I will accept late turn-ins, with 5 point per day of lateness, up to 5 days. **However, no turn-ins will be accepted more than 5 days late or after the last day of class (Friday, December 3, 2021 at 11:59pm).**
2. **Canvas exercises (15% of your grade):** I will publish a few exercises on Canvas every week (they can be found under the Assignments tab). Each exercise will either consist of an online programming problem, 1-5 questions based on the lecture material, or a Canvas discussion as a

response to a prompt. Discussion questions must be completed by the assigned due date. Please only rely on your texts, lecture notes, and computer (Canvas, Eclipse, and built in Java documentation) to answer the exercises; please **do not** search the Internet for the answer or ask anyone for the answer (again, see Academic Dishonesty). **Late submissions are generally not accepted for exercises.**

3. **Team project (10% of your grade**): You will participate in a team to define, specify, implement, test and document a project of greater complexity than a single programmer could be expected to do within a specified time frame. Be thinking about what interests and challenges you as you proceed with the course. 40% of the project will be devoted to documents that define and specify your project and 50% will be the implementation. The final 10% will come from submission of weekly meeting minutes, which will begin when groups are assigned. **You will be expected to demonstrate and discuss your project in an online meeting during the last week of class, followed by submitting documentation and code on Canvas.**

4. **Final exam (25% of your grade).** We will have an on-campus final exam in building 15, room 201. The exam will be held on Tuesday, December 7 from 11:30am-1:30pm. Please see the "Face-to-Face Testing" section above.

Grading

Grades will be posted to Canvas as soon as assignments are graded. If you think there has been an error, contact me as soon as possible to have it corrected. If you need assistance accessing your Canvas account, please contact eLearning here: https://www.tacomacc.edu/academics-programs/academic-support/elearning/. Final grades will be assigned using the following scale:

A: [94, 100]%	A-: [90, 94)%	B+: [87, 90)%	B: [84, 87)%	B-: [80, 84)%
C+: [77, 80)%	C: [74, 77)%	C-: [70, 74)%	D: [61, 70)%	E: below 61%

Note that in the above scale, a "[" or "]" means that the value is included in the range. A "(" or ")" means that the value is excluded from the range.

Academic Dishonesty

This gets a little complicated in a programming course. Each of the three kinds of homework have slightly different policies. For the Canvas exercises, you should answer the questions yourself without searching for answers online or asking any people for a solution. You should only need to reference the Canvas class material and Oracle website. Please contact me if the feedback given by the online environment is confusing. I will try to respond within 24 hours but may not have time to respond before the deadline if you contact me the day the exercise is due!

For the team project, groups are expected to work together and share information, and may also contribute expertise to other teams. In addition, you may want to search on sites such as Stack Overflow for the answers to **focused questions about how to accomplish small steps of your solution, not the solution itself**. In other words, "How do I accomplish this small step in Java". Any code should be adapted to your specific situation, paraphrased, and retyped, not copied and pasted verbatim. You should only adapt **one statement of code at a time** and understand the meaning of all the code you adapt.

For the weekly programming assignments, I expect you to discuss concepts with other students. However, I expect your submissions to be your own. Never receive code (copy and paste, reading over the shoulder, email, download, and so forth) from other people who know about the specifics of your

assignment or who implemented a similar project. Never show or send your code to other students. You should not look at other students' code at all. The only reason to look at another student's code is if you are helping that student fix a specific problem or bug in their code that they asked you about first – focus on the specific problem at hand, don't show them your code, and don't copy their code. The assignments are designed so that you should never have to search online for answers, other than the class material found on Canvas (textbooks, handouts, etc.) and Java documentation found at the Oracle website (including the Javadoc and Java tutorial). These resources explain and demonstrate concepts rather than give away code. Use of Stack Overflow may be problematic; on the one hand, it could contain the demonstration for a general concept you could apply, or it could contain code that you would be tempted to copy verbatim. I recommend staying away from this site for the assignments. **Cheating websites like Chegg are prohibited.** It is easy for me to launch an investigation on Chegg (and similar websites) to determine if code has been copied from there.

I use **JPlag**, a system developed at Karlsruhe University in Germany that finds similarities among multiple sets of source code files to detect software plagiarism. **JPlag** does not merely compare bytes of text, but is aware of programming language syntax and program structure and hence is robust against many kinds of attempts to disguise similarities between plagiarized files.

If you are working with other students, and aren't sure whether the "help" you are getting crosses the threshold of copying – you may contact me, and I'll run a **JPlag** comparison for you. I'll need their code and yours. I use 90% code similarity as a threshold of unacceptable assignments. You **may** get one warning at my discretion for borderline cases.

Finally, don't try to "fool" **JPlag**. Simple tricks like renaming variables, changing indenting and line breaks, rearranging sections of code, they are all understood and expected by the authors. Trying to fool the program is just stronger evidence of intent to deceive.

Students who are conscientious never have difficulty with this system. It is only students who don't understand programming, and scramble to get too much help to turn in late assignments who get in trouble.

As stated in the TCC catalog, 'Students are expected to be honest and forthright in their academic endeavors. Cheating, plagiarism, fabrication or other forms of academic dishonesty corrupt the learning process and threaten the educational environment for all students." The complete Administrative Process for Academic Dishonesty is available on the TCC website at: http://www.tacomacc.edu/abouttcc/policies/administrativeprocedureforacademicdishonesty/

I will not condone cheating or other forms of academic dishonesty. Cheating, and behaviors reasonably associated with cheating, are a violation of the student disciplinary code and are punishable through the disciplinary procedures of the college. **If I find persuasive evidence that you have been cheating, I reserve the right to fail you in the entire course.** Instructors have an obligation to report incidents of cheating, or suspected cheating, in accordance with the procedures outlined in the Student Rights and Responsibilities Code.

Graded Item Submission Guidelines
Assignments will be submitted on Canvas. Some assignments can be completed directly in Canvas. Others will require uploading a file from your computer to Canvas.

Student Resources

During this quarter, you may encounter a need that goes beyond this class that affects your academic performance. We are here to help you be successful. You have access to great resources on campus and people ready to support you. Login to Canvas and access the Student Resources course to learn about some of these excellent resources.

Equity, Diversity, and Inclusion

I acknowledge that oppression (i.e. racism, sexism, classism, heterosexism) exists and impacts nearly every part of our society. One of the mechanisms of oppression is that we are all consciously and unconsciously taught misinformation about our own groups and/or groups that are traditionally normalized and marginalized. I accept responsibility for not just repudiating these myths and stereotypes but combating them as well.

I agree to never demean, devalue, or in any way belittle people for their experiences and opinions. At the same time, I recognize that my experience is not everyone's experience. It is my responsibility to create a safe atmosphere for open discussion and to call out inappropriate and harmful behavior

Access and Accommodations

Your access in the learning environment is important to me. If you have already established disability accommodations with the Access Services office, please communicate your approved accommodations to me at your earliest convenience so we can discuss your needs in this course. You can request that Access Services email your Letter of Accommodation to me, or you can provide me with printed copies.

If you have a disability or health condition that may benefit from accommodations to ensure access and support success in this course—and have *not yet* established services, please contact Access Services at (253) 460-4437 or access@tacomacc.edu.

Access Services offers resources and coordinates reasonable accommodations for students with disabilities and/or temporary health conditions. Services are established through an interactive process that begins with an intake appointment. Access Services is located in Building 7.

Reasonable Accommodations for Religion/Conscience

Students who will be absent from course activities due to reasons of faith or conscience may seek reasonable accommodations so that grades are not impacted. Such requests must be made within the first two weeks of the quarter and should follow the procedures listed in the Leave for Faith & Conscience policy webpage.

Safety

We care about the safety of our campus and community and all of us are needed in order to create a safe and secure learning environment. Please review the full list of safety resources available to you, which are listed on our TCC Ready website. Additionally, please watch this brief video to familiarize yourself with our general safety practices. Please **program the Public Safety number into your cell phone (253-566-5111)** or 5111 from any campus phone (If this number is inoperable due to an emergency, dial **253-495-4146**) and sign up for TCC Alerts! to receive emergency notifications to your cell phone.

Changes to the syllabus

Changes and clarifications may be made to this syllabus. Whenever a change is made, it will be announced on Canvas and in class. We will make sure to discuss the change as a class to make sure that it is fair and make sense.