

CS 149: Introduction to Programming – Sections 8 and 9
James Madison University Fall 2021 (3 credits)

Come Learn How to Program

Have you ever wanted to create your own game or application on a computer or cell phone? We all use computer applications everyday, including mobile, web-based, tablet-based, and those that run on laptop/desktop systems. Heck, even my thermostat in my house now connects to the Internet. This class provides the foundational skills for building computer applications using the **Java** programming language. Java is a very powerful and popular language and is utilized in many enterprise level applications, web applications, and mobile applications.

This class will utilize team-based learning and allow you develop fundamental programming skills that will be useful not only if you want to pursue studies in computer science, but in almost every discipline. Some of the programming assignments will be completed in class in a team environment, allowing you to develop not only great programming skills, both also communication and team building skills that are sought after by **all** employers. Programming can be challenging, but it is also very rewarding.

I hope you will join me for this class that will put you on the path to becoming a great computer scientist.

1 Basic Course Information

1.1 Meeting Times, Locations, and Instructors/Teaching Assistants

Section	Days	Time	Location
8	T/R	9:40 - 10:55	ISAT/CS 143 (Tuesday) & ISAT/CS 248 (Thursday)
9	T/R	11:20 - 12:35	ISAT/CS 143 (Tuesday) & ISAT/CS 248 (Thursday)



Name	Role	Section(s)	Email
Dr. Kevin Molloy	Instructor	8 & 9	molloykp@jmu.edu
Chris Settles	Teaching Assistant	8	settl2cr@dukes.jmu.edu
Maddie Hince	Teaching Assistant	9	hinceme@dukes.jmu.edu

1.2 Office Hours

My office is located in the **ISAT/CS building room 216**. My office hours for this semester are:

- Tuesday 5:30 pm - 6:30 pm
- Wednesday 2:00 pm - 4:30 pm
- Friday 10:00 am- 11:30 am (sometimes virtual, link to Zoom is on Canvas)

No appointment is required to come to office hours. However, If you are not able to meet with me during these times, you can email me (molloykp@jmu.edu) to schedule a meeting.

1.3 Course Technologies

The following online resources will be used in this class.



Canvas will be used for making all announcements and posting assignments (reading, programming, quizzes). The modules section will contain links and organize all required work for the class. <https://canvas.jmu.edu/courses/1816991>



Piazza is a place to ask questions about the class and get feedback from myself, the TAs, and other students. This allows your answers to be viewed/shared by your colleagues/classmates. <https://piazza.com/jmu/fall2021/cs149>



Autolab will be used to submit almost all of your programming assignments. <https://autolab.cs.jmu.edu>



All written assignments (including exams) will be returned to you via this website. It allows me to provide detailed feedback on your responses and for you to inquiry about your marks on a specific question. <https://www.gradescope.com>

1.4 Prerequisites

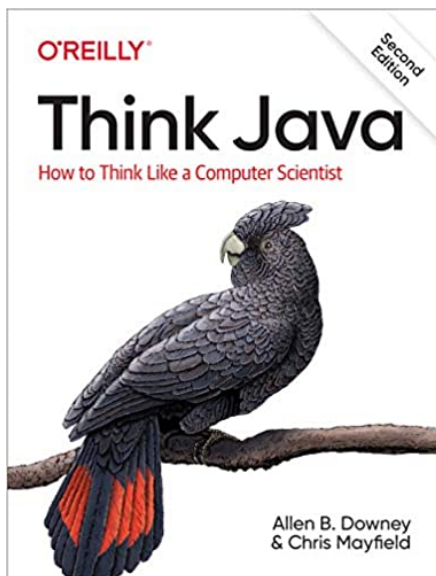
You must have completed **MATH 155**, **MATH 156**, or obtained a sufficient score on the **Mathematics Placement Exam**.

1.5 Course Description and Goals

Official course description: Fundamental problem-solving techniques using a modern programming language. Topics include variables, input/output, decisions, loops, functions, arrays, and objects. Students learn about algorithm development, testing strategies, and software tools. By the end of this course, you should be able to:

- Explain fundamental programming concepts (e.g, variables, methods, decisions, loops, arrays, objects) using appropriate terminology.
- Describe basic elements of high-level programming languages, including expressions, statements, functions, modules, and libraries.
- Read and interpret software specifications and write source code from them.
- Use automated software tools and processes to test your programs thoroughly.
- Distinguish “appropriate collaboration” from academic dishonesty/cheating on assignments and exams.
- Evaluate your own work for compliance with requirements and style guidelines.
- Develop correct and efficient algorithms to solve problems using computation.

1.6 Required Texts



Think Java: How to Think Like a Computer Scientist, 2nd Ed. by Allen Downey and [Chris Mayfield](https://greenteapress.com/wp/think-java-2e/). Green Tea Press, Needham, MA. <https://greenteapress.com/wp/think-java-2e/>.

The pdf is available for **free online**. This course aligns with this open-source textbook. If you don’t like the book, you are certainly welcome to use additional resources, including other textbooks.

An interactive version of the textbook, which allows you to compile and work through some of the coding examples in this textbook, is available at <https://books.trinket.io/thinkjava2/>. Where applicable, reading assignments will provide links to both the textbook and the interactive version (commonly referred to as the **Trinket** version).

Another excellent resource for beginning programming is [Dr. Bernstein’s](#) book “[Patterns for Beginning Programmers](#)”, which is available for free.

1.7 Computing Resources

You will require access to a computer for this class. A machine running Windows, Mac OS, or Ubuntu/Linux will work fine. My opinion is that Mac OS and Ubuntu/Linux machines are easier to use. Information is available in our Canvas class on how to utilize a virtual machine that will supply all the software you will need for this class (and provide a common environment to what you will experience in the labs).

1.8 Expectations/Keys to Success

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday
HW due 11 pm		POGIL Activity HW Prob 1	Reading (45 mins) Formative Concept Quiz HW Prob 1 and 2 due Muddiest Points	Mini-Lecture Labs/In-class work In-class assessment quiz Work on HW	HW prob 3-5

Homework

In a three-hour course, you should expect **six hours** of homework per week. How you manage your schedule is up to you, but do spend some time each day on this course. The above schedule is a suggestion, which you might notice does not require any work on the **weekends**.

POGIL Activity

This class utilizes **Process-Oriented Guided Inquiry Learning**, or POGIL. These activities, typically conducted on Tuesday, will involve group and active learning. This method has been shown to be highly effective, which is the reason I employ it in this course.

You will be assigned into groups of between 3 and 5 students, and each assigned roles that rotate. Groups will be randomly assigned for the first few weeks. We will utilize room 143 for this activity, so, plan to go there for most Tuesday classes. Three programming assignments will be due in the later part of the semester. They are designed to take about 8 hours to complete. The key to completing these assignments is to start earlier (ask anyone who has taken CS 149).

Seeking Help

Thanks to the feedback of the students who have come before you and a very dedicate faculty, this course has a great support network. This network includes teaching assistants (see section 2), peer led review sessions (see Section 2.1), and **Piazza** (an online forum built to facilitate discussions).

Generally, I am responsive to questions on Piazza during the week, but not very responsive questions over the weekend. If you choose to complete assignments at the last minute or after the deadline, especially after the first PA, you have significantly decreased your chances of successful completion. Piazza is configured so that I get an email when a question is posted, so emailing me is not a quicker way of reaching me. Plus by posting to Piazza, you will have a chance of being answered

by a classmate, TA. Email should be reserved for questions whose answers would only benefit you personally or only I would know the answer to.

1.9 Communication

We will use a number of different tools for communication in this course. These include:

- **Canvas** is our central repository for information. You can configure canvas to email/alert you when new information is posted.
- Use public posts on **Piazza** to discuss the material related to this course.
- **Canvas** and **Autolab** will be used to submit assignments and disseminate grades
- **EMail the professor** (molloykp@jmu.edu) if you have logistic or personal issues to discuss such as setting up an appointment outside of office hours, if a health problem arises, or if you have a personal emergency.
- **Office Hours** No appointments are required to attend office hours or you can make an appointment with me.

1.10 Attendance and Participation

Regular attendance and fully engaged participation is expected. Your grade will be partially based on in-class activities, so, attendance will affect your grade.

Please silence your cell phone while class is in session. If you have a laptop or tablet, you are encouraged to bring it to class and use it to work along with programming examples and exercises. Mute the volume to avoid unintended interruptions, and do not use any electronic devices for activities that may distract other students. Repeated violations of this policy may result in disciplinary action or a grade penalty in the course.

2 Teaching Assistants and Review Sessions

Undergraduate teaching assistants are generally available in the labs, and their availability is posted on the **CS Wiki** (<https://wiki.cs.jmu.edu/department/cs-success/start>).

Do not rely on them to fix your code, as you won't have access to the TAs during exams. However, I do encourage you to seek help from the TAs and/or come to my office hours for clarification on concepts.

2.1 Review Sessions – The 4th hour

Together with the teaching assistants, the department sponsors a review session each week called **The 4th Hour**. This review session covers the material from the prior week of class. These sessions are lead by teaching assistants, many of which have taken CS 149 and can understand/relate to the challenges and rewards of taking this class.

2.2 The Role of a Teaching Assistant

When available, the TA's job is to help you learn the course material. Lab assistants should **not fix your code** and/or write code for you. They will *try* and answer your questions of a general nature, *try* to help you find your mistakes, and *try* to help you interpret error messages and output. To some extent, they will also *try* to help you understand assignments.

2.3 Prerequisites for Getting Help

The lab assistants can't provide effective help unless you can both read code and refer to it using the appropriate terminology. Hence, before you ask a question of the lab assistants, you must try and ask it using the correct terminology, discuss the question with the TA, and understand the answer. For example, asking a question about something in the textbook is OK. Asking questions that are clearly answered in the textbook (potentially because you skipped/did not complete the reading) is not acceptable.

A lab assistant who decides that you have not asked a question properly or that you are not prepared to understand the answer, may politely inform you that she/he can't help. If this happens to you, please be respectful of the TA, as they have been trained to not simply provide answers, but rather, to help you obtain the answers yourself.

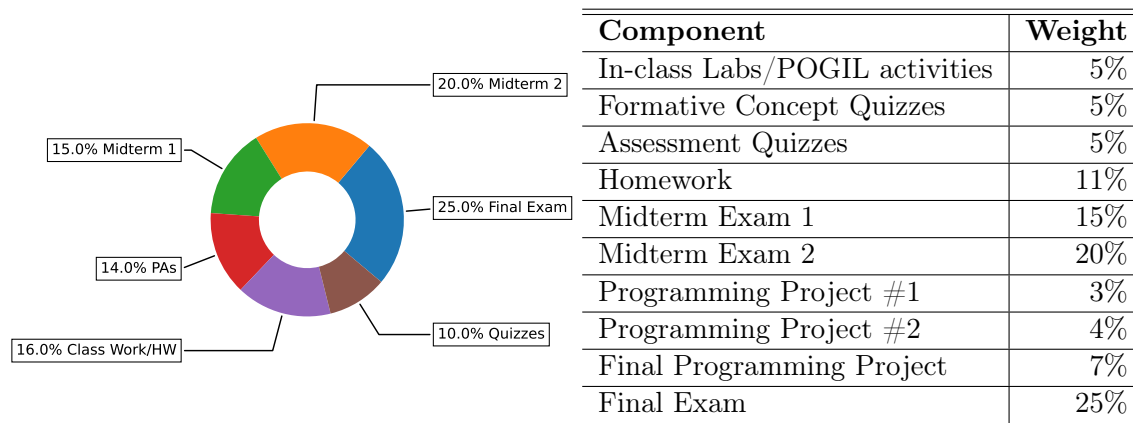
2.4 Your Responsibilities

Remember that lab assistants are students. In fact, they are students that only recently completed the course that you are taking. They sometimes make mistakes, give bad advice, and point you in the wrong direction. They are trying to help, but may not be able to. Sometimes, they even make things worse.

You are ultimately responsible for all of the assignments in this course. The lab assistants are in no way responsible for your work. They provide advice, you must decide whether you should heed their advice or not. Also, you must be sure not to ask them questions that would violate the Honor Code.

3 Methods of Evaluation and Grading Policies

You are responsible for all material discussed in lecture, assigned readings, announcements on Canvas, and posted videos. Your final grade will be determined according to the following percentages:



Letter grades will be assigned on the scale A=90-100, B=80-89, C=70-79, D=60-69, F=0-59, with potential minor adjustments after considering the overall performance of the class and actual distribution of numeric scores. I will use + and - grades at my discretion. **Students who do not earn 60% of the points on the final exam will receive a letter grade no higher than C+ for the course.**

3.1 Formative Concept Quizzes

Each week you will have an online quiz (typically due Wednesday evening). I will drop your lowest quiz grade at the end of the semester to account for a “bad day” or unforeseen circumstances that prohibit you from submitting a particular quiz. This quiz should be taken after completing the work from Tuesday’s class and any assigned reading/posted video lectures.

3.2 Assessment Quizzes

On most weeks, you will have an in class assessment quiz at the end of Thursday’s class. This quiz will include both written and coding questions. Its purpose is to help you evaluate your readiness for higher stake assessments (exams). You will have the option of retaking a quiz on this material if you perform any one of the following activities following the quiz:

- attend and **participate** in the 4th hour review sessions the week following the quiz
- attend office hours within 6 days of taking the quiz
- schedule a meeting with me outside office hours to discuss your quiz (again, within 6 days of taking the quiz)

Data from prior semesters show a **strong** correlation between final scores on the assessment quizzes and scores on the exam.

3.3 Midterms/Final

This course will have two in-class midterms exams and a comprehensive exam during finals week. You must be present or give a valid doctors excuse or similar note confirming your absence if you miss any exam.

If you must be absent during an exam for a legitimate reason, you must contact me **at least one week beforehand** to make special arrangements. Failure to make prior arrangements for a missed exam will result in a zero grade. Except in extraordinary situations, you will not be excused from exams. Your instructor will be the sole arbiter of whether a situation qualifies as extraordinary. Hence, you should behave as if you will not be excused from exams for any reason.

3.4 Missed and Late Assignment Policy

It will **not** be possible to receive credit for in-class work that is missed due to absence. In recognition of the fact that absences are occasionally unavoidable, I will drop your lowest score for a single in-class activities. There will be no late acceptance of any formative concept quizzes, however, I will also drop the lowest formative concept quiz score for the semester.

Homework assignments that are due on Wednesday will NOT be accepted late (since we will review the answers in class on Thursday). Homework assignments that are due on Sunday and programming assignments (PAs) will be accepted up to 24 hours late with a penalty of 10%. In extreme, documented circumstances (e.g., hospitalization), I will make reasonable accommodations.

3.5 Grading Disputes

If you believe I have made an error while grading your work or calculating your final score, please bring it to my attention after class or during office hours. If I determine that there has been a simple mistake, I will fix it immediately and no formal request is necessary.

If you believe an exam question or assignment has been graded unfairly, you must submit a written formal request for a regrade via email. Such requests must be submitted within one week of when the assignment in question is returned to you. **Any coursework submitted for reconsideration may be regraded in its entirety, which could result in a lower score if warranted.**

4 Course Policies

Important announcements will be made in class and/or via canvas. Please make it a habit to check your email daily during the semester.

Although every effort has been made to be complete and accurate, unforeseen circumstances arising during the semester could require the adjustment of any material given here. Consequently, given due notice to students, I reserve the right to change any information on this syllabus or in other course materials.

You are permitted to use course materials for your own personal use only. Course materials may not be distributed publicly or provided to others (except to other students currently enrolled in the course), in any way or format unless explicitly allowed.

4.1 Important Notes

You must achieve a B- or better grade to continue on to CS 159. See the note in Section 3 on the final exam requirements with respect to achieving a B- or better grade in the course.

4.2 Adding and Dropping the Course

Students are responsible for adding and dropping the course and verifying these actions in MyMadison. The last date to drop this class with a "W" grade is October 27th. Please consult the appropriate [registrar dates](#) for other deadlines. I will not give “WP” or “WF” grades to students requesting a drop after the deadline except in extraordinary circumstances.

4.3 Disability Accommodations

If you need an accommodation based on the impact of a disability, you must contact the [Office of Disability Services](#) if you have not previously done so. Disability Services will provide you with an Access Plan letter that will verify your need for services and make recommendations for accommodations to be used in the classroom. Once you have shown me this letter, we will sit down and review the course requirements, your disability characteristics, and your requested accommodations to develop an individualized plan appropriate for this course. I will not make any accommodations without the appropriate documentation, as I am not qualified to diagnose disabilities.

4.4 Excused Absences

Besides the policies in this syllabus, the University’s policies apply during the semester. Various policies that may be relevant appear in the Undergraduate Catalog.

If you must be absent during an exam for a legitimate reason, you must contact me at least one week beforehand to make arrangements. Failure to make prior arrangements for a missed exam will result in a zero grade. Excused absences will be granted at my discretion and only with appropriate documentation. Please contact me as soon as possible if you wish to request an excused absence.

Missing an exam for reasons such as illness, religious observance, participation in required university activities, or family or personal emergency (such as a serious automobile accident or the funeral of a close relative) all are circumstances that *may* qualify as an excused absence. Where possible you should attempt by all means necessary to attend and take exams at their regularly scheduled class period.

Observance of religious events will be accommodated for students of any faith.

4.5 Classroom Behavior

Students are expected to maintain a high level of civility for all participants in and out of class meetings. This includes respecting the beliefs of participants of all genders, ethnicities, and social backgrounds. Harassment of any type will not be tolerated and failure to behave in a respectful manner will result in referrals to University Counseling or the Office of Student Judicial Affairs. Any instances of sexual harassment will be reported to the Office of Equal Opportunity according to the following policy:

<https://www.jmu.edu/JMUpolicy/policies/1340.shtml>

4.6 Inclement Weather

This class will operate in accord with JMU's inclement weather policy available at <http://www.jmu.edu/JMUpolicy/1309.shtml>.

4.7 Religious Observation Accommodations

I will give reasonable and appropriate accommodations to students requesting them on grounds of religious observation. If you require such accommodations you must notify me at least two weeks in advance.

5 Academic Honesty and Collaboration

5.1 Academic Honesty

You are expected to comply with the JMU Honor Code as stated in the Student Handbook on all work in this class (see the [Honor Council website](#)).

Consulting with other students about problems and solutions is not necessarily a violation of the honor code, depending on the particular assignment. All final work turned in for an assignment must be your own unless it is a group project. You must list **all** sources of assistance at the top of your assignments (teaching assistants, colleagues/class mates, members of a group in a group project). You may not share source or binary code on programming assignments unless the project specification explicitly allows it. If you are in doubt about whether something is an honor code violation, please contact me.

If I find evidence of a violation of the honor code, I will bring the matter to the attention of the involved individuals via email and request a face-to-face meeting. As per section IV of the honor code, first time student offenders may agree that a violation has occurred and accept an appropriate penalty by submitting an "Informal Resolution Agreement Form" to the honor council. If the student is not a first-time offender or if there is disagreement about the violation or penalty, the matter will be referred to the honor council under section V of the honor code.

5.2 PRIME DIRECTIVE

PRIME DIRECTIVE: Be able to explain your own work including homework code and exam solutions.

Nearly all cheating in programming can be averted by adhering to the PRIME-DIRECTIVE. Students may be asked at any time to explain code or exam solutions they submit. Inability to do so will be construed as evidence of misconduct. More specific guidelines are given below.

Thou Shalt Not

For the purposes of this course, the following actions constitute scholastic misconduct (cheating):

- Directly copying someone else's solution to a homework problem, including student solutions from a previous semester
- Directly copying an answer from some outside source such as the Internet or friend for a homework problem
- Making use of an Instructor Solution manual to complete homework problems
- Paying someone for a homework solution or submitting someone else's work as your own
- Posting solutions to any web site including our course web site
- Collaborating or copying someone else's answer during an exam
- Aiding or abetting any of the above
- Witnessing any of the above and failing to report it the instructor immediately

Fair Collaboration

The purpose of this course is to learn about programming and learning from one another is a great help. To that end, the following actions **will NOT be considered cheating in this course**.

- Talking to other students in the course about HW problems and informally describing how a problem may be solved.
- Getting or giving help fixing a bug or two: a second set of eyes is a great boon to finding that misplaced semicolon that is preventing your code from compiling.
- Searching the Internet for alternative presentations of a programming concept.
- When unsure whether collaboration is fair or not, stop the activity until it can be cleared with instructor.

A key component of academic integrity is giving credit where credit is due. If you receive assistance, either from another student or from some other source, you must provide a clear and explicit acknowledgment of that assistance in your submission.

Acknowledgments

Many members of JMU's computer science faculty have contributed material to this course, of the course materials we will be using this semester, including portions of this syllabus. Contributors include: David Bernstein, John Bowers, Alvin Chao, Chris Mayfield, Nathan Sprague and Dee Weikle.