Computer Science 1321
Java for Programmers
Spring 2026

Class Time: Half Term Jan 20 -March 13th Mon 3-4:40pm
Location: Carson Energy Center 0121
Instructor: Dr. Sean Mackay Email: snmackay@ou.edu Office: DEH 243

This course has three essential learning objectives:

1. Learning to write a few lines of code to solve given problems.
2. Learning to write complete computer programs to solve given problems.
3. Developing conceptual understanding of how computer programs work.

Students achieve these objectives through extensive practice in the Java programming language. This practice occurs in phases. First, students read about the lecture material before class. Then we discuss the material in class while working many examples together and checking understanding with periodic short quizzes via Canvas in lecture. After class, students practice with projects and homework assignments. This means that there are class deadlines every two or three days. Repeated practice leads to mastery.

Office Hours:

- Monday @ 12:10-2pm (starting week 2) in DEH 243

- Wednesday @ 2-4:30pm (starting week 2) in DEH 243

Class Structure: This class is a synchronous in person class. You are expected to attend all classes on time and obtain approved absences when you are unable to make it due to illness or other excused absences. You must have a working laptop that can stay on for 2 hours for the class. Participation represents an extremely large percentage of your grade, and thus attendance is essential for your success.

Canvas Learning Management System: https://canvas.ou.edu
Log in with your OUNetID (usually the first 4 letters of your last name followed by a 4-digit number). All assignments, deadlines, grades, announcements, and course documents will be posted to the CS 1321 Canvas page for your section. It is your responsibility to regularly check for updates. You can configure Canvas to email you notifications or send them through text messages. Please do this so you don't miss assignments.

Prerequisites:

1. Math 1503 (College Algebra) or placement in higher mathematics courses.
2. Basic computer literacy such as the ability to install software and navigate folder structures.
3. Prior programming experience in another language. If you do not have substantial programming experience, then taking CS 1323/4 is advised. If you are unsure, speak with me as soon as possible either in person after the first class or via email.

Free Tutoring: In addition to faculty and TA office hours, the School of Computer Science through the William Kerber Foundation Teaching Scholars, and the Dean's Leadership Council (DLC) of the Gallogly College of Engineering. These tutors and I should be the first place you go for help, since we are all trained specifically to support students in this course. The times, people, locations, and Zoom links will appear on Canvas under Pages as they are available (usually about 2-3 weeks into the semester).

Topics Covered: Programs, Java, identifiers, variables, assignment statements, constants, memory diagrams, primitive data types, operations on primitive data, conditional statements, input and output, repetition, methods, parameters, arguments, return values, passing by value, passing by sharing, nested control statements, one dimensional arrays, objects, user defined classes, and classes from the Java Application Programmers Interface (API) (including Arrays, ArrayList, Character, Collections, Double, Integer, Float, Math, Scanner, String, and StringBuilder), tracing program execution, debugging.

Learning Outcomes:

1 Select the appropriate type for data stored in a program from int, double, String, char, boolean, and identify constants.
2 Document code using comments, indentation, whitespace, selection of appropriate identifiers, Javadoc (reading, not writing), and standard Java code conventions.
3 Trace memory allocation of primitive data, arrays, and String and ArrayList objects in a single method.
4 Construct and trace statements using arithmetic, relational, and logic and assignment operators on data, considering both precedence and promotion.
5 Call class methods the Math, Integer, Character, Double, String, Arrays, and Collections classes.
6 Perform input from the console and from files using the Scanner class.
7 Perform output using the PrintStream (System.out) and PrintWriter classes
8 Construct and trace conditional and loop statements.
9 Select and nest control structures (sequence, conditional statements, loops, and methods) to meet requirements.
10 Construct and call methods to meet requirements when given the signature using parameter passing and return values.
11 Implement classes with static methods and no instance or class data to solve programming problems.
12 Predict the contents of memory following method calls using pass by sharing (reference), pass by value, and return values with primivite data, arrays, and String and ArrayList objects.
13 Solve programming problems using arrays of primitive data types and String.

14 Design method signatures that allow arrays that are entirely and partially filled with data to manipulated.
15 Trace selection and insertion sort on arrays of primitive data types and String.
16 Call instance methods in the ArrayList, String, Integer, Character, Double classes.
17 Solve programming problems using ArrayList of wrapper classes (Integer, Character, Double) and String.
18 Construct UML diagrams from API documentation and code.
19 Implement and call methods from simple student defined classes with instance and class data, accessors, and mutators from UML diagrams.
20 Find and fix errors in student written programs.

ABET: Students will increase their ability to meet the following ABET outcomes:
Outcome 1: Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.
Outcome 2: Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

Required Materials: Please purchase the following items **as soon as possible**. Each is required to complete a different type of assignment and counts for your grade. If you are unable to afford these items at the start of the semester, let me know so I can excuse your work for a little while or help you find another way to get access.

1. Laptop computer with network access for class and lab.
   - You are responsible for having a working laptop available for every class and laboratory. This includes finding a replacement with the necessary software installed if your laptop requires repairs during the semester. It's battery should be able to last for the duration of the class.
2. Zyante online textbook (ZyBook) with labs (zyLabs).
   - Click on the first Zybook assignment in Canvas, which has the title "Chapter 2".
   - Click the link at the bottom of the assignment page to open the Zyante website in a new window.
   - Subscribe to the book.

Final Exam: We will have a final exam during the last class time. The exam will be open book open note.

Study Advice: In technical fields like computer science, the only way to master the subject matter is to practice. Learning to program is like learning to play a musical instrument. You can read a hundred books on playing the piano, but if you don't sit down in front of a keyboard and practice, you won't be able to produce any interesting music. Similarly, if you only attend lectures or read the textbook, you're going to find it difficult to write functioning computer programs of any complexity.

To facilitate your practice, this course has different types of assignments, which are described below. Start each assignment as early as you can and get help from me, a teaching

assistant, or a tutor if you get stuck. Programming can be challenging, which is why we have so many office hours. If you work hard on and understand the assignments, you should do well on the exams, earn a good grade in the class, and generally have a rewarding semester.

Assignments: This course has 5 different assignment types. Each is designed to help you learn the material in a different way. All assignments are due at 11:59 PM on their posted due dates.

1. Zyante (zyBook): The online textbook will introduce you to new topics before I cover them in class.
   - Each section contains activities to reinforce the ideas in the text. Activities come in two types: participation and challenge. **You are only required to complete participation activities**, although you are welcome to complete challenge activities for extra practice.
   - Each question can be attempted an unlimited number of times without a penalty. You earn 1 point for each question answered correctly before the deadline.
   - Some sections are marked as optional. You are not required to complete these sections.
2. Participation: I will be ensuring you are participating in class. Attendance is mandatory, and will be tracked using Tophat. Should you fail to participate in the work assigned that day, you may forfeit part or all your attendance for the day regardless of whether you have completed the Tophat.
3. Projects: These assignments will require you to work on the code both in class and out of class. You will work in groups of 1-2. We will discuss this structure during the first day of class.
   - Projects are assigned each Monday and are generally due on Zylab the following Monday.

Grading: Your final course grade is calculated using the weights in the table below. The weights applied to the Zyante grades are intentionally low. This allows you to learn from mistakes with only small penalties. Completing these assignments is how most students develop the conceptual understanding needed to do well on the homework, projects, and especially exams.

Assignment	Weight (%)
Zyante	30
Participation	15
Projects (8 total)	30
Final Exam	25

* Zyante questions are worth 1 point.

Letter Grades: Your course grade will be converted into a letter using a scale no higher than the following. The scale may be lowered at the end of the semester at my discretion.

Letter	Percentage
A	90+
B	80–89
C	70–79
D	60–69
F	Below 60

Borderline Grades: It would be nice if all course grades fell cleanly into the ranges shown above. Most semesters, however, a handful of letter grades are decided by only a few points. In these difficult cases, I will use the following algorithm: A course grade is considered a borderline grade if it is within three points of the next higher letter. For example, 87, 68 and 79 are borderline course grades, but 81 and 92 are not. For borderline grades, if the grade on the final exam is above the threshold for the higher letter, the higher letter will be given. Otherwise, the lower letter will be given.

Grade Checking: Canvas has a grade book that stores the raw data used to calculate your course grade. It is your responsibility to periodically check that your grades are recorded properly. If you find an error, email me as soon as possible, and I will correct it. **The grade summary on Canvas is not accurate and cannot be made accurate** since Canvas does not allow the implementation of several course policies. In addition, Canvas sometimes excuses grades that are not actually excused. Treat Canvas grades with great suspicion and recalculate them by hand using the rules in this syllabus if you need an accurate calculation.

Electronic Devices in Class: You may use laptops, tablets, cell phones, and other electronic devices in class in ways that enhance your learning. These devices may not be used in ways that distract other students (e.g., playing games, texting friends, watching videos, or making noise). Your cell phone should generally be off during class time unless you are using it for class-related work or have a good reason to have it on (e.g. family responsibilities). My cell phone is on every day because I have family responsibilities.

Academic Integrity: The Student's Guide to Academic Integrity defines academic misconduct as "any act that improperly affects the evaluation of a student's academic performance or achievement," including cheating on exams, improper collaboration on assignments, and plagiarism (https://www.ou.edu/integrity/students).

The most common violation in this course is plagiarism, usually on homework and projects. Plagiarism is "an act or instance of using or closely imitating the language and thoughts of another author and the representation of that author's work as one's own" (https://www.dictionary.com/browse/plagiarism). When completing assignments in this class, please keep the following in mind:

- Solutions should not be copied from internet sources, including websites, artificial intelligence software (like ChatGPT), and paid professional programmers.

- Do not show, give, or email another student a copy of your work before the submission deadline. Do not trust other students to not plagiarize your work.
- Do not use artificial intelligence systems like ChatGPT to complete your work. Copying from an automated source is plagiarism, just like copying from a human and for exactly the same reasons.
- The penalties for permitting your work to be copied are usually the same as the penalties for copying someone else's work, since it is not possible to distinguish the person who copied from the person who allowed their work to be copied. If I can determine who created the work, the person copying the work will receive a harsher penalty.

Projects are done in the class with one-two assigned partners. Sharing your code with this person(s) is not only permitted but required. Sharing code with anyone else may be a violation of academic integrity.

Proper and Improper Collaboration: When you pass this class with a grade of C or better, I am certifying that you are a competent Java programmer. I cannot make this certification without seeing work that you complete on your own. Interactive programming tutors, homework, and examinations should be the work of a single individual, not their friends, not an artificial intelligence system, and not their tutor. It is permissible and even advisable to talk to other students in the class for help completing or improving your work, however, this help must not interfere with my ability to evaluate the quality and quantity of your understanding of computer programming. To understand this distinction, please review the examples in the table below. This is not a comprehensive list of all the ways in which academic integrity can or cannot be violated.

Situation	**Integrity Violation?**
Students A and B meet and work on their homework together. Neither student prepared anything in advance.	**Yes**
Students A and B create drafts of their homework assignment independently and meet to compare answers and discuss their understanding of the material. Each student decides independently whether to make changes that are discussed.	No, and this is a great idea
Students A and B agree to prepare drafts of their homework assignment independently, but only Student A does. Student A shares his draft with Student B who reviews it and offers suggestions for improvement.	**Yes**
Students A and B agree that student A will work the even problems and student B will work the odd problems. They share their work.	**Yes**
Student A has completed a project and is helping student B complete the same project. Student A explains to student B what student B's code does, which is different than what student B thinks the code does. Student B determines how to modify the code independently.	No
Student A has completed a project and is helping student B complete the same project. Student B is having trouble getting one part of the program to work, so student A texts student B three lines of their solution.	**Yes**

Student A has completed a project and is helping student B complete the same project. Student B is having difficulty getting the program to work, so student A tells student B exactly what to type for several lines.	**Yes**
Student A has completed a project and is helping student B complete the same project. Student B is having difficulty getting the program to work, so student A suggests that student B use a specific debugging strategy (e.g., "Trace these lines of the code").	No
Student A has completed a project and is helping student B complete the same project. Student A shows student B an example program in the online textbook that will be helpful in figuring out the solution to the problem.	No
Students A and B work on a project together but are not laboratory partners. After they have finished it, student A takes the code and modifies it so the programs do not appear to be identical.*	**Yes**

*I can easily detect these changes using automated software. Don't bother doing this.

Chegg and Other Online Tutoring Sources: There are a wide variety of tutoring resources available through paid websites. Many of these sites have students upload assignments and solutions and surreptitiously provide these documents to other students. What appears to be a session with a tutor may be, behind the scenes, the tutor doing a search of their company database of solutions to share. By using these sites you risk being charged with academic misconduct, either by supplying other students with answers they did not author or by receiving someone else's answer that you did not author. Since these companies are not open with students about their practices, you cannot know whether a tutor is providing meaningful support (for example, identifying misunderstandings of content and explaining them like our teaching assistants and I would) or simply feeding you someone else's solution. The tutor's actions can result in different students submitting answers that are identical, which may be flagged as academic misconduct during grading, especially when the solution is incorrect (which it often is since their tutors may lack expertise). See the table below for specific examples. **There is no way to use these sites without risking being charged, and even committing, academic misconduct at this time.**

Scenario	**May Be Charged With Academic Misconduct?**	**Guilty of Academic Misconduct**
You use the website to receive help. During the process, the assignment and all or part of your solution are stored on a company computer. The assignment or solution are subsequently delivered by the company to another student that you do not know.	Yes, because another student may submit an assignment that is identical to yours.	No, but you're going to have to prove it since two students with identical solutions is usually considered strong evidence of academic misconduct, especially if the solution is not correct.
You use the website to receive help. The assignment has already been uploaded, and your tutor provides you	Yes, because you committed academic misconduct.	Yes, and you may not even be aware that the tutor was providing someone else's solution instead of teaching.

with the uploaded solution. You submit all or part of this solution as your work.		This is academic misconduct because you are copying from a tutor.

Academic Integrity Process: Upon the first documented occurrence of academic misconduct, I will report the violation to the Office of Academic Integrity Programs. If you are found guilty by this process, the penalty may be a failing grade in the class and being suspended from college for a semester. If you have committed academic misconduct previously, the sanctions can be more severe, including expulsion from OU. The grade sanction that I usually request for plagiarism on a single assignment is a zero on the assignment. The grade sanction that I usually request for a student accused of cheating on an examination is an F in the course. The procedure to be followed is documented in the University of Oklahoma Academic Misconduct Code. If I elect to admonish you, the appeals process is described here: https://www.ou.edu/integrity/students. The highest penalty for an admonition is a zero on the assignment.

Tutors and Academic Integrity: Before you hire a private tutor, please take advantage of the many people who are paid by OU to support this class. These people are trained to tutor properly. Private tutors can be a source of support if you are struggling in the class, but only if the tutor is aware of the distinction between teaching you the material so that you can do your own work and completing assignments for you. Tutors who simply complete your assignments are not only failing to help you learn, they are also abetting academic misconduct. Each of the situations listed above in the table of collaboration scenarios applies when student A is a tutor.

Generative AI Policy: This course focuses on how people learn to produce, understand, and fix computer programs. While AI is becoming more skilled at creating computer programs, there is no evidence that allowing students to use AI to complete programming assignments develops the necessary conceptual understanding of programming to allow students to succeed as computer scientists. At this time, synthesizing code from memory is a method that has been proven to create successful computer scientists. Therefore, using AI to perform assignments will be permitted in this class only when specified. There are ways to demonstrate that code was generated by AI and these methods are rapidly improving. Do not assume that copying from AI cannot be detected or proven.

Students who copy generated AI results into assignments are committing plagiarism, just as if they had copied from another student.

- Use of generative AI tools is only allowed when the assignment specifically permits it. If the assignment is silent on the use of generative AI tools, they are not allowed.
- If there is a question about whether a student completed work independently or with the use of these tools, Dr. Mackay will invite the student to her office to explain the assignment solution in detail. Students who choose not to come within 48 hours or who cannot explain the work they submitted will be charged with an academic integrity violation.

Schedule: Check Canvas for due dates and what assignments or work is coming up. This course is somewhat self-paced and I will adjust based on how you all are meeting the material. All dates on Canvas are subject to change, though any changes will be made in your favor (e.g. additional time).

Week 1 -> Getting familiar with Java, Project 1 (NOTE: no class week 1, this may be done beginning of week 2 or you may begin in week 1 async).

-> Storing Primitive Data

-> Arithmetic Operators

-> User Interaction

-> Conditional Statements

Week 2 -> Project 2

-> Logical Operators

-> While Statements

-> Methods Part 1

Week 3 -> Project 3

-> Methods Part 2

-> For Loops

-> Arrays and Primitive Data

Week 4 -> Project 4

-> Methods and References

-> Perfect/Oversize Arrays

-> Nested Loops

Week 5 -> Project 5

-> Sorting

-> Using Objects from the API

-> Classes With Generics

Week 6 -> Project 6

-> Building News Classes

-> Access Modification

Week 7 -> Project 7

-> Aggregation

Week 8 -> Project 8

-> Final Exam

University Policies

Mental Health Support Services

Support is available for any student experiencing mental health issues that are impacting their academic success. Students can either been seen at the University Counseling Center (UCC) located on the second floor of Goddard Health Center or receive 24/7/365 crisis support from a licensed mental health provider through TimelyCare. To schedule an appointment or receive more information about mental health resources at OU please call the UCC at 405-325-2911 or visit University Counseling Center. The UCC is located at 620 Elm Ave., Room 201, Norman, OK 73019.

Title IX Resources and Reporting Requirement

The University of Oklahoma faculty are committed to creating a safe learning environment for all members of our community, free from sex-based discrimination, including sexual harassment, domestic and dating violence, sexual assault, and stalking, in accordance with Title IX. There are resources available to those impacted, including: speaking with someone confidentially about your options, medical attention, counseling, reporting, academic support, and safety plans. If you have (or someone you know has) experienced any form of sex-based discrimination or violence and wish to speak with someone confidentially, please contact OU Advocates (available 24/7 at 405-615-0013) or University Counseling Center (M-F 8 a.m. to 5 p.m. at 405-325-2911).

Because the University of Oklahoma is committed to the safety of you and other students, and because of our Title IX obligations, I, as well as other faculty, Graduate Assistants, and Teaching Assistants, are mandatory reporters. This means that we are obligated to report sex-based violence that has been disclosed to us to the Institutional Equity Office. This means that we are obligated to report sex-based violence that has been disclosed to us to the Institutional Equity Office. This includes disclosures that occur in: class discussion, writing assignments, discussion boards, emails and during Student/Office Hours. You may also choose to report directly to the Institutional Equity Office. After a report is filed, the Title IX Coordinator will reach out to provide resources, support, and information and the reported information will remain private. For more information regarding the University's Title IX Grievance procedures, reporting, or support measures, please visit Institutional Equity Office at 405-325-3546.

Adjustments for Pregnancy and Related Issues

Should you need modifications or adjustments to your course requirements because of pregnancy or a pregnancy-related condition, please request modifications via the Institutional Equity Office website or

call the Institutional Equity Office at 405/325-3546 as soon as possible. Also, see the Institutional Equity Office FAQ on Pregnant and Parenting Students' Rights for answers to commonly asked questions.

Reasonable Accommodation Policy

The University of Oklahoma (OU) is committed to the goal of achieving equal educational opportunity and full educational participation for students with disabilities. If you have already established reasonable accommodations with the Accessibility and Disability Resource Center (ADRC), please log into iAdvise to request your semester accommodations as soon as possible and contact me privately, so that we have adequate time to arrange your approved academic accommodations.

If you have not yet established services through ADRC, but have a documented disability and require accommodations, please complete ADRC's pre-registration form to begin the registration process. ADRC facilitates the interactive process that establishes reasonable accommodations for students at OU. For more information on ADRC registration procedures, please review their website. You may also contact them at (405)325-3852 or adrc@ou.edu, or visit www.ou.edu/adrc for more information.

Note: disabilities may include, but are not limited to, mental health, chronic health, physical, vision, hearing, learning and attention disabilities, pregnancy-related. ADRC can also support students experiencing temporary medical conditions.

Religious Observance

It is the policy of the University to excuse the absences of students that result from religious observances and to reschedule examinations and additional required classwork that may fall on religious holidays, without penalty. [See Faculty Handbook 3.15.2]

Final Exam Preparation Period

Pre-finals week will be defined as the seven calendar days before the first day of finals. Faculty may cover new course material throughout this week. For specific provisions of the policy please refer to OU's Final Exam Preparation Period policy.

Emergency Protocol

During an emergency, there are official university procedures that will maximize your safety.

Severe Weather: If you receive an OU Alert to seek refuge or hear a tornado siren that signals severe weather.

1. Look for severe weather refuge location maps located inside most OU buildings near the entrances.
2. Seek refuge inside a building. Do not leave one building to seek shelter in another building that you deem safer. If outside, get into the nearest building.

3. Go to the building's severe weather refuge location. If you do not know where that is, go to the lowest level possible and seek refuge in an innermost room. Avoid outside doors and windows.
4. Get in, Get Down, Cover Up
5. Wait for official notice to resume normal activities.

Additional Weather Safety Information is available through the Department of Campus Safety.

The University of Oklahoma Active Threat Guidance

The University of Oklahoma embraces a Run, Hide, Fight strategy for active threats on campus. This strategy is well known, widely accepted, and proven to save lives. To receive emergency campus alerts, be sure to update your contact information and preferences in the account settings section at one.ou.edu.

RUN: Running away from the threat is usually the best option. If it is safe to run, run as far away from the threat as possible. Call 911 when you are in a safe location and let them know from which OU campus you're calling from and location of active threat.

HIDE: If running is not practical, the next best option is to hide. Lock and barricade all doors; turn of all lights; turn down your phone's volume; search for improvised weapons; hide behind solid objects and walls; and hide yourself completely and stay quiet. Remain in place until law enforcement arrives. Be patient and remain hidden.

FIGHT: If you are unable to run or hide, the last best option is to fight. Have one or more improvised weapons with you and be prepared to attack. Attack them when they are least expecting it and hit them where it hurts most: the face (specifically eyes, nose, and ears), the throat, the diaphragm (solar plexus), and the groin.

Please save OUPD's contact information in your phone.

NORMAN campus: *For non-emergencies call (405) 325-1717. For emergencies call (405) 325-1911 or dial 911.*

TULSA campus: *For non-emergencies call (918) 660-3900. For emergencies call (918) 660-3333 or dial 911.*

Fire Alarm/General Emergency

If you receive an OU Alert that there is danger inside or near the building, or the fire alarm inside the building activates:

1. *LEAVE* the building. Do not use the elevators.
2. *KNOW* at least two building exits
3. *ASSIST* those that may need help
4. *PROCEED* to the emergency assembly area

5. *ONCE safely outside, NOTIFY first responders of anyone that may still be inside building due to mobility issues.*
6. *WAIT* for official notice before attempting to re-enter the building.

OU Fire Safety on Campus