Syllabus

CS 2100 - Data Structures and Algorithms 1 – Spring 2024

— Instructors

Instructor (Email)	Office Hours	Office Hour Location
Briana Morrison (bbmorrison @virginia.edu)	➢ Tuesdays: 3:00pm - 4:00pm[1] ➢ Wednesdays: 3:00pm – 4:00pm ➢ Other times: *by appointment* (https://bit.ly/morrisonAppt)	Rice 407
Derrick Stone (djs6d@virginia.edu)	➢ Mondays: 2:00 - 3:00 pm	Rice 209

Instructor Office Hours begin on Tuesday, January 23, 2024.

— Course Information

Meeting dates: January 16, 2024 to April 30, 2024

Lecture:

Section	Days & Time	Location	Professor
001	M/W/F, 12:00-12:50pm	Chemistry Bldg 402	Briana Morrison
002	M/W/F, 11:00-11:50am	Olsson 120	Derrick Stone

Lab: *(all lab sections are held on Monday)*

Section	Day & Time	Location
101	Monday, 3:00-4:45pm	Olsson 120
102	Monday, 5:00-6:45pm	Rice Hall 130
103	Monday, 7:00-7:45pm	Rice Hall 130
104	Monday, 3:30-5:15pm	Mechanical Engr 205

The first day of lab begins on Monday, January 22, 2024. For students who add the class late, this lab may be made up on your own – instructions can be found within Canvas.

— Learning Management System: Canvas

This course will use an accompanying course site in UVACanvas.
Log in to UVACanvas using your NetBadge credentials. (Need help with NetBadge?)
Getting Started:

- New to Canvas? Learn the basics in this 3.5-minute tutorial video.

[1] ***All times in this Syllabus are given in Eastern Time (ET)***

- A mobile app is available. Visit the App Store or Google Play and search for Canvas Student. To connect your app to UVA Canvas, tap "Find my School" and search for "University of Virginia (UVA)." With the Canvas mobile app, you can quickly view grades and announcements and send messages. *Please ***do not use*** the mobile app for taking quizzes and tests.
- Know how to get UVACanvas help.
    - Log in to UVACanvas and click Help in the main navigation
    - Search the Canvas Guides (Step-by-step instructions)
    - 24/7 Chat with Canvas Support
    - 24/7 Canvas Support Hotline: +1 (866) 897-5086

— Communication

Welcome your questions, comments, and concerns. The entire course staff is here to help you succeed in the course. Please know that with over 600 students enrolled in the course and only 2 instructors, it is impossible for us to answer the deluge of emails from students in a timely manner without some rules. This semester we will be using a ticketing system to handle all requests from students. ***DO NOT EMAIL the professor OR TAs directly***. Send all requests to CS2100@cshelpdesk.atlassian.net (JIRA). You will receive a ticket number and a member of the instructional staff should respond within 24 hours (except weekends).

TA names, emails, and their office hours will be posted on the course **Calendar** (which is linked to via Canvas); all **Office Hours begin** on Tuesday, January 23, 2024.

— Important Dates

- First day of CS 2100: January 17, 2024 (Wednesday)
- Add deadline: January 31, 2024
- Drop deadline (without "W"): February 24, 2024
- Spring Break: Marcy 2-10, 2024 (*no class*)
- Drop deadline (with "W"): March 13, 2024
- Last day of CS 2100: April 29, 2024 (Courses end: April 30, 2024)
- Exam Reading Days: May 5 and May 8, 2024
- CS 2100 Final Exam [all sections]: ***May 2, 2024***; starting at **7:00pm**
  (Please plan accordingly! We will only accommodate direct exam conflicts.)

— Textbooks / Resources

There will be no recommended nor required textbooks this semester. All content that you need will be on the course website. We are currently in the process of curating appropriate resources to make available to students. If you find a good resource, please let us know!
You may, however, like to have access to a good Java reference book. If you don't already have one, please try an online source such as:

- Java Programming (a wikibook)
- "Data Structures with Java" by John R. Hubbard
- Thinking in Java by Bruce Eckel
- Introduction to Computer Science using Java by Bradley Kjell
- Fred Swartz's topic-based review.
- Oracle's Java documentation *(highly recommended!)*

Since our course focuses on becoming a good programmer in Java, our goal is to help you learn all you can about Java programming. Resist the temptation to copy and paste code from answer-providing sites such as Stack Overflow or others. In addition to violating the collaboration and honor policy (see below), copy/pasting code interferes with the learning process of what constitutes good, efficient code.

We also suggest places to practice your coding skills. Practice makes perfect! Please see the Resources Page for sites that will allow you to practice your coding skills. Doing one or two practice problems every day will improve your coding skills and solidify your understanding.

— Assessments (How you will be graded)

The course is roughly broken up into three sections: 1) Learning Java, 2) Intro to Data Structures, and 3) Sorting. Here you can find the full **Course Schedule**.
Below is the breakdown of how your final grade will be calculated:

Item	# in the semester	Points per assessment	Total Points
Quiz Part As	5	35	175
Quiz Part Bs	5	65	325
Programming HW	13	30	360
Labs	9	15	130
Syllabus Quiz	1	10	10
			1000

You may drop one HW assignment.
Lab 1 is only worth 10 points

Letter grades will be assigned according to the following letter grade mapping, **with no rounding**:

Grade	**A+**	**A**	**A-**	**B+**	**B**	**B-**	**C+**	**C**	**C-**	**D+**	**D**	**D-**	**F**
Lower Bound	98.0	93.0	90.0	87.0	83.0	80.0	77.0	73.0	70.0	67.0	63.0	60.0	0

— Course Flow

Generally, here is what a typical week will look like:

Monday	**Tuesday**	**Wednesday**	**Thurs**	**Friday**	**Sat**	**Sun**
Lecture Lab Meeting - Lab Asgn - Quiz Part B Quiz Part A Closes		Lecture HW Due	HW Released	Lecture Quiz Part A Opens		

You will attend *lectures* on Mondays, Wednesdays, and Fridays. *Programming assignments* will be released on Thursday mornings and will be due the following Wednesday at 11:59pm. Mondays you will have lab meetings. For most lab meetings you will have a *lab assignment*, designed to be completed in pairs and to be completed during the lab session. Every three weeks (or so) there will be a *Quiz*.

— Syllabus Quiz

Every student must complete the **Syllabus Quiz**. This quiz is open to all students for a little over two weeks (from Jan 17 through Feb 1 at 11:59pm). Every student must take the quiz (or be removed from the course). This quiz is **taken on Canvas** and can be **attempted as many times** as necessary (so earn a **100% score**). You should see the score you earned on Canvas Gradebook.

— Exams / Quizzes

Quizzes are broken into two parts:
Part A: An individual at-home, timed, coding assignment. You will be given the specifications for a piece of code which you are to write or debug. You will submit it to GradeScope for auto-testing and receive immediate feedback. These will be open book and open notes using your IDE. You may also be required to answer questions explaining your logic for your code or comparing your code to another piece of code. Part A will open around 3 pm on Friday and close at 11:59pm on Monday. You will have up to 2 hours to complete it – generally plenty of time without taking up your entire weekend. There is also some manual grading of the code. Regrade requests will be available once all student quizzes have been graded and are available through Gradescope.
Part B: An individual quiz to be completed during the lab session. This consists of more typical exam questions: multiple choice, fill-in-the-blank, short answer, etc. to be completed on your computer and are closed-book with no additional resources. Regrade requests are available once quiz scores have been released and are available through a Google form, the link made available through a Canvas announcement.

There will be 5 quizzes in total for the semester; 5 part A and 5 part B.

— Final Exam

For the final exam, you will be allowed to retake up to **four (4)** parts of any quiz. This means you can retake Quiz 1A, Quiz 2A, Quiz 3A and Quiz 4A if you want. Or retake Quiz 1A, Quiz 2B, Quiz 3A, and Quiz 4B. In other words, $\sum \text{Part A} + \sum \text{Part B} = 4$. This is a way for you to **improve your score on any quizzes that you want** (we take the higher score)**.** Or you may choose to not retake any parts of any quizzes!

The retake for Quiz Part Bs will occur during the common course final exam time of Tuesday, May 2, 7-10pm. The retake for Quiz Part As is TBD. The quizzes will open in the morning and must be submitted by 9:00pm that evening. Note that each part will be timed – A parts will be up to 2 hours and B parts will be 50-90 minutes each.

— Homework Submissions

Programming assignments (homework) will open on Thursday mornings and are due the following Wednesday at 11:59pm. Most of the time, these involve writing code, implementing a data structure, etc. Homework assignments are submitted online on **Gradescope** and are automatically tested. You may submit your homeworks an unlimited number of times, unless otherwise specified in the assignment instructions.

Early Policy: Programming assignments may be submitted early for bonus points! Assignments that compile cleanly and pass at least 50% of the automated test cases will be eligible for bonus points. Assignments with their final submission at least 48 hours before the due date will receive an additional 3 points. Those with their final submission at least 24 hours before the due date will receive an additional 1 point.

Late Policy: Homework assignments may be submitted up to 2 days late, with 5% penalty (1.5 points) for each late day. No assignments will be accepted after ***Friday at 5pm*** of the week the assignment is due. Because assignments are weekly, being more than 2 days late on an assignment puts you 2 days behind in starting the next assignment.

The programming homework assignment with the lowest score will be dropped from your final grade calculation (i.e., you can drop one programming homework assignment). After the drop, we will waive one late homework penalty, adding back up to 3 points to your homework grade.

This late policy is designed to handle illness, travel issues, small computer issues, etc. *You are encouraged to start your assignments early.*

If you have extenuating circumstances requiring you to need additional time, please submit a JIRA ticket – only serious illness or other situations will be considered.

— Lab Submissions

Attending lab is worth 5 points. Completing the lab assignment is worth 10 points. You should have your work checked off for completion by the lab TAs. If you don't finish, or if you miss lab one week, you have one week to complete the assigned work. You may go to TA Office Hours or show the TAs the following week the completed work; however, you will only be awarded 10 points if you did not attend lab. For illness or university sponsored absences, please submit a ticket to JIRA.

For lab assignments, if you work in pairs, only 1 person needs to submit, but please ensure that both names and computing IDs are located in the file.

— Regrade Policy

After assessments are graded and returned, it may be that you would like to discuss the points you earned for one or more questions. This is called a *regrade request*. Your request must be specific and address an issue when an assignment may have been graded incorrectly. All regrade requests must be made **within 7 days** of the assessment being returned to the student. If you do not show understanding of the material in your request, it will be denied. We reserve the right to re-grade the entire assessment, not just the specific request, so it is possible that your grade may go up or down after the regrade request. Please see the following table for where to request a regrade:

Assessment	**Regrade Request Platform**	**Notes**
Programming HW	Gradescope	After grades have been published, you will receive an email from Gradescope indicating regrade requests are open.
Quiz Part A	Gradescope	After grades have been published, you will receive an email from Gradescope indicating regrade requests are open.
Quiz Part B	Google Form	After grades have been published, there will be a Canvas announcement with a link to the Google form.
Lab	JIRA ticket	

— Computing Resources

There are some specific computer resources needed for this course:

- You need to have a working computer that can run IntelliJ IDEA IDE (https://www.jetbrains.com/idea/). Any modern laptop or desktop will have no problem running this software.

- The *first lab assignment* involves a short tutorial of downloading, installing, and using the **IntelliJ** environment. Follow the provided instructions to ensure you have the proper setup on your laptop. The TAs will be available to help you during the lab. Note that you are free to use any Java IDE that you wish but instructor examples and labs will be done using IntelliJ.

— Honesty and the Honor Policy

The goal of this course is for each learner to achieve the learning objectives. As a general rule, unless otherwise stated, programming assignments and quizzes must be the result of **individual academic effort**. Labs can be completed in pairs, and only one person of the pair needs to submit.

No plagiarism (nor anything like it)
You **must** cite any and every source you consult, other than those explicitly provided by the course itself. If you work with, obtain, or receive help from another source (Internet website, tutor, online video, etc.), the source should be cited as a comment in code on the assignment. NOTHING should be copied *or retyped into the submitted solution;* no electronic versions of any assignment or quiz should be shared with anyone. Any copied work is an Honor Code violation. If you are in doubt, please ask!

Write your own code (within the collaboration policy)
When the collaboration policy states that work must represent individual effort, you must write your own code. Not just type it (though you need to do that too): **compose it yourself**, as your own original work.

We ask you to program to help you learn the content covered in the programming assignment and to help you demonstrate to us your knowledge. This is *unlike* industry, where you program to create a product. Because it is your mind we are looking to help develop and measure, it is your mind that must do all the work. Working with others is *not* OK.

Our TAs have been trained to provide help that does not undermine the primary purpose of helping you learn. Other people (tutors, fellow students, etc.) have not. As such, *you should be careful receiving help from those who are not TAs for the course.*

Collaboration Policy
We understand that occasionally you may get stuck on something or need a little help over a bump. In and during lab time you are free to talk with any and all other classmates about programming assignments (NOT Quizzes). You are free to look at someone else's code and help them debug it. There should be NO pictures of code (in fact, phones should be completely away during all labs). We are hoping that by allowing this time of discussion you will receive all the help you need the day before the assignment is due.

Understanding what you submit
Your understanding is the primary deliverable of our assignments, not the code itself. As such, we may ask you to explain aspects of a solution you turn in and may dock points if it appears you simply copied someone else's ideas (or just guessed a lot of things until one worked) without understanding them.

*Note that trying to take a shortcut to homework solutions **only harms you in the long run**. Know that if you intend to pursue a career in computer science, you will be expected to know how to solve problems similar to*

those you will see in this class. Skipping the learning process now merely means that you will need to master these skills later.

Penalties: up to 100% course grade penalty. If course staff detect cheating, plagiarism, sharing, copying another solution to an assignment or quiz (including portions thereof), or other dishonest behavior and honor code infractions, they may impose any penalty up to and including a failing grade (F) in the course. *This is independent of, and in addition to, the operations of the Honor Code.*

- **1st Offense:** You will receive **zero (0)** points on that assessment
- **2nd Offense: Automatic F** in the course

A note regarding Honor Code Referrals: The course is not bound by the decisions of the Honor Committee as it relates to expulsion. If we refer you for an honor violation, and you are not expelled, the above penalties will still be applied.

— Generative AI tools in This Course

Generative artificial intelligence tools—software that creates new text, images, computer code, audio, video, and other content—have become widely available. Well-known examples include ChatGPT for text and code, CoPilot for code, and DALL•E for images. This policy governs all such tools, including those released during our semester together. You may use generative AI tools on assignments in this course when explicitly permitted. Each graded assessment will indicate whether or not Generative AI tools can be used. If you do use generative AI tools on assignments in this class, you must properly document and credit the tools themselves. Cite the tool you used, just as you would cite a website; usually these are listed in the "Resources" comments at the beginning of submitted programs. Additionally, please include a brief description of how you used the tool. If you choose to use generative AI tools, please remember that they are typically trained on limited datasets that may be out of date. Additionally, generative AI datasets are trained on pre-existing material, including copyrighted material; therefore, relying on a generative AI tool may result in plagiarism or copyright violations. Finally, keep in mind that the goal of generative AI tools is to produce content that seems to have been produced by a human, not to produce accurate or reliable content; therefore, relying on a generative AI tool may result in your submission of inaccurate content. It is your responsibility—not the tool's—to assure the quality, integrity, and accuracy of work you submit in any college course. If you use generative AI tools to complete assignments in this course, in ways that have not been explicitly authorized, you may be subject to UVA and course Academic Integrity policies. In addition, you must be wary of unintentional plagiarism or fabrication of data. Please act with integrity, for the sake of both your personal character and your academic record.[2] Please see the resource provided within Canvas for more information.

— Disclaimer

This syllabus is to be considered a reference document that can and will be adjusted through the course of the semester to address changing needs. This syllabus can be changed at any time without notification. It is up to the student to monitor this page for any changes. Final authority on any decision in this course rests with the instructors, not with this document.

[2] Generated using https://courses.pepperdine.edu/access/content/user/cheard/Twine/Generative_AI_Syllabus_Statement.html