

Section Info

Section 51/51H - CRN 27556 / 28181
Tuesday/Thursday 5:30 pm - 7:50 pm W310

This course has an honors option for Honor College students. Please meet with the professor by the end of the first week of class if you are interested in taking the class for your honors course requirements.

Course Modality: in the classroom

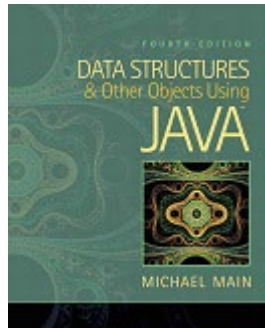
The class will be conducted as a regular class with Lecture and Lab time in the classroom. Canvas is the official source for: Syllabus, Class Schedule, Slides, Demos, Homework/Lab/Project Assignments, Exam Information

Syllabus Versions

- Version 0.0 - 1/13/2024 - First-draft

Book Information

Data Structures & Other Objects Using Java, Michael Main, 4th Edition



Picture of book:

Instructor Information:

Name: Stephen T. Brower
Work # (908) 526-1200 x8259

Office: West Building W324
preferred email: stephen.brower@raritanval.edu

Department Information:

Math and Computer Science Department. Chair: Dr. Lori Austin, Lori.Austin@raritanval.edu

Spring 2024 Student Hours (1/17-4/30):

Location: West 324 and Zoom (see Canvas for Zoom links)

- Monday 1:00 - 2:30 & 5:30 - 6:30
- Wednesday 1:00 - 2:30 & 5:30 - 6:30
- and by appointment

Course Overview

Prerequisite: CSIT(CISY) 105 Foundations of Computer Science or GDEV(CISY) 242 Object Oriented Programming:

This course introduces students to the fundamental data structures used in Computer Science. The data structures covered include linked lists, doubly linked lists, stacks, queues, trees, and graphs. Algorithms that manipulate these data structures are discussed and used in laboratory work. Students are introduced to the run-time analysis of algorithms and basic algorithms for searching and sorting. The Honors Option is available for this course.

General Education Goals:

After completion of this course, students will be able to:

1. design and develop data structures that efficiently address program requirements (GE-NJ 2, GE-NJ 4)
2. analyze the data structures used in computer applications and the issues surrounding their implementation (GE-NJ 2, GE-NJ 4)
3. apply quantitative reasoning to analyze the performance of data structure algorithms in order to efficiently solve problems (GE-NJ 2)

Course learning outcomes:

At the conclusion of the course, students will be able to:

1. compare and contrast the basic data structures used in Computer Science: lists, stacks, queues, trees and graphs
2. identify and implement the basic operations for manipulating each type of data structure
3. create data structures using Java
4. analyze the run-time analysis of algorithms and express them using $O ()$ notation
5. apply recursion to data structure operations
6. identify the appropriate data structure for a given problem
7. analyze algorithms to search or sort the data in various data structures (arrays, queues, stacks, etc.) and interpret their run-time performance
8. create and execute test plans which include the testing of boundary conditions

Honors Option

This course has an honors option for Honor College students. Please meet with the professor by the end of the first week of class if you are interested in taking the class for your honors course requirements.

An Honors Option is available for this course. A general requirement for Honors courses and Honors Option courses is a minimum grade point average of 3.5

"Definition: Students pursuing the Honors Option will be required to demonstrate a higher level of knowledge and skill in each of their course programming projects. They will be required to

take the concepts introduced and generalize them for broader application. Students will also do independent work researching application programming interfaces." (Master Course Outline)

For this Honors Option, the demonstration of a higher level of knowledge and skill will be through enhanced Labs and Projects. The independent work for research will be through additional research components for Homework

The expectation is that all of the 'honors option' exercises be completed. The expectation is that 90% of the 'honors option' exercises should be completed. If more than 10% of the 'honors option' exercises are not completed then for the assignments completed without the honors option beyond 10% of the assignments a penalty will be assessed

For some of the assignments, there are extra credit options for the students in the 'regular section' that are mandatory for the students in the 'honors section' and the grade is based on the proportion of the total points. For example: suppose there is 15 points extra credit on an assignment, for the students in the 'honors section' their grade is based on the total points / 115. If a student switches from the 'honors section' back to the 'regular section', the grade will be recalculated.

Enrolling in the Honors Option Section: By the end of the first week (1/23) fill out an Add/Drop form that drops Section 01 and adds Section 01H and get the instructor's signature (by 1/23) and the department chair's signature. After the form is filled out and signed then take it to the registrar.

Exiting the Honors Option Section: If you wish to switch back to the regular section, by the withdrawal date (4/5), fill out an Add/Drop form that drops Section 01H and adds Section 01 and get the instructor's signature and the department chair. After the form is filled out and signed then take it to the registrar.

Course Management, Structure and Pace

This course meets for approximately five hours a week for 14 weeks. Students should plan on putting in at least two hours of study time for every hour spent in class for lecture. Additionally, students should plan study time of another half hour for every hour spent in the lab. Students who are successful in this class typically spend approximately seven hours *outside of class* each week working on the subject. This includes reviewing class notes, reading and studying the textbooks, doing homework and reviewing demo Java code **including "testers" for labs.**

Attending all lectures is essential for success in this course because this reinforces and explains the material presented in the textbook. Additional programming techniques, which may not be found in the book, will be demonstrated in class and during lab time.

Grade Determinants:

Item	Percent
Homework *	5%
Data Structure Labs *	5%
Programming Labs *	10%
Programming Projects	10%
Midterm Exam (2/29)	20%
Final Exam (5/2) **	50%

* At the end of the semester the lowest Homework grade, the lowest Data Structure Lab grade, and the lowest Programming Lab grade will be dropped

**** In order to pass the course, you will need an overall average of 60 in the course and you will need a 60 or above on the Final Exam.**

Grade % Range

A	89.5-100.0+
B+	86.5-89.4
B	79.5-86.4
C+	76.5-79.4
C	69.5 - 76.4
D	59.5 - 69.4
F	0 - 59.4

Software/Computer Requirements:

Although there is lab time incorporated into the class to work on “labs”, there may not be enough time to work on the “projects” during class time. You *may* need access to a computer with Java outside of class.

If you own a computer, you can download the Java JDK and NetBeans which will replicate the setup that we have on campus. There will be links from the Course Homepage in Canvas with some information.

It is not a requirement to use NetBeans. You can use Eclipse, JGrasp, BlueJay, Notepad++, some other IDE or even Notepad, but you may need to create a ‘project’ in your IDE and import all of the .java files, and you may need to adjust the package statements in the files if only a NetBeans project is provided.

If you do not own a computer, you can use the Open Lab. See the hours posted outside the Open Lab

Note: If we become displaced from campus, the Open Lab may not be available or have reduced hours

Laptops in the classroom:

You are allowed to use your own laptops in the classroom. Wireless devices should be able to connect to "RVCC-Student" by providing your G# and password. (see: <https://commons.raritanval.edu/admin/mis/tshome/TSGuide/Pages/Network.aspx>)

Which Email to use and Email Response Time

If you have a question or have an issue submitting an assignment in Canvas, the preferred (fastest) way to contact the instructor is via his preferred email: stephen.brower@raritanval.edu

Over the last several semesters, I found the email system embedded in Canvas frustrating, and the Canvas alerts are not always punctual.

So please email me at: stephen.brower@raritanval.edu

The goal is to respond in less than 24 hours. Please don't expect a response after 10 pm.

Occasionally there are known exceptions such as around Spring Break where a response may take a little longer. If the instructor knows ahead of time there will be a period of unavailability longer than 24 hours, that will be communicated to the class.

Course Routine

Classes will usually consist of 3 possible segments: Lecture, Lab Lecture, and Lab Time

Lecture

A majority of the lectures will be on topics within Data Structures. Some of the lecture will be the theoretical nature of Data Structures, which will include a number of crudely drawn pictures, a discussion of algorithms that act on Data Structures, and in some cases Pseudocode.

Lab Lecture

The Lab Lecture will be an introduction to the week's lab which could be a review of the topic(s) introduced in the Lecture or simply "here".

Lab

There are two kinds of labs: *Data Structure Labs* and *Programming Labs*. *Data Structure Labs* will be paper-based group exercises related to the Data Structure topics from the lectures. *Programming Lab* will be for individual work using the computer to implement Data Structures in Java to solve problems. During Lab time, take as many breaks as needed.

If you finish the lab early, you should look ahead on the Class Schedule to work on the next programming project (if posted) or do the next homework that is due

Questions during lab time must be asked in person during lab time. Emails sent during lab time will be ignored.

Pattern (mostly)

Not every week follows a pattern but many weeks will have Lecture and Data Structure Lab on Tuesday and Programming Lab time on Thursdays

Homework

The Homework will be posted in Canvas. Homework must be submitted electronically via Canvas as a document saved as a **.docx** or **.rtf** file.:

Once an assignment is due it is closed in Canvas and work cannot be submitted through Canvas. Attaching files to the “comments” is not allowed and those files will be ignored.

See below for the Late Policy(don't be late) and the Cheating Policy (don't cheat)

Data Structure Labs

For classes that introduce new Data Structures, the Data Structure Lab on a topic may be a drawing showing a series of operations on a Data Structure.

The instructor will review the Data Structure Lab before the beginning of Lab time.

The Data Structure Labs will be distributed on paper in class and you will return the paper. These Data Structure Labs can be done in small groups (2-3 per group)

It is preferable that you submit the Data Structure Labs by the end of the Data Structure Lab time, but the latest to submit is the end of class (4:50pm)

See below for the Late Policy (don't be late). Since the Data Structure Labs are group-based, the Cheating Policy as stated below isn't fully applicable.

Programming Labs

Programming Labs will be programs written independently in Java to implement a Data Structure to solve some problems.

All of the Programming Labs have some assumptions for clean input so that the focus can be on the writing and use of the data structure. For example assuming a number entered is a number.

Some Programming Labs have been abbreviated to be done in the class-time allotted and those abbreviated labs are due at the end of the class by 4:50 pm.

Some Programming Labs are distributed one week and are due the following week.

The instructor will review the Programming Lab before the beginning of lab time.

Questions during lab time must be asked in person during lab time. Emails sent during lab time will be ignored.

The .java* files are to be submitted as attachments to the assignment in Canvas - *alternatively, a .zip file of the folder that contains the .java files or a .zip of the NetBeans project.

Once an assignment is due it is closed in Canvas and work cannot be submitted through Canvas. Attaching files to the “comments” is not allowed and those files will be ignored.

See below for the Late Policy (don't be late) and the Cheating Policy (don't cheat)

Programming Lab Compiling Policy

As a second course using Java, the expectation is that your Java code will compile. For all of the Programming Labs from Programming Lab 4 and beyond, if the code submitted does not compile, or if the instructor's tester used to test the classes submitted does not compile, the assignment is a 0.

The only edits that will be made to get the program to compile is to remove the -# that Canvas adds to the file name when there are additional submissions and the package statement

Programming Projects

The idea behind Programming Projects is that they are more complex than labs and are used to demonstrate mastery of Data Structures. A number of them also do not have the assumptions for clean input that the Programming Labs have.

A reminder that Java is in the open Lab in the West Building which is open 6 days a week. Java is also available for download from the Oracle Website. See the instructor's website for more information.

The .java* files are to be submitted as attachments to the assignment in Canvas. alternatively, a .zip file of the folder that contains the .java files or a .zip of the NetBeans project.

Once an assignment is due it is closed in Canvas and work cannot be submitted through Canvas. Attaching files to the “comments” is not allowed and those files will be ignored.

See below for the Late Policy (don't be late) and the Cheating Policy (don't cheat)

Project Compiling Policy

As a second course using Java, the expectation is that your Java code will compile. For all of the Projects, if the code submitted does not compile, or if the instructor's tester used to test the classes submitted does not compile, the assignment is a 0.

The only edits that will be made to get the program to compile is to remove the -# that Canvas adds to the file name when there are additional submissions and the package statement

Exams

Note the Class Schedule for the dates of the exams. If you are late for the exam, you will only have the time until the scheduled end of the exam.

Exams must be taken on days assigned. If you know ahead of time that you cannot take an exam, ask the instructor to arrange for the exam to be left in the testing center or to arrange another time.

About a week before each exam, a 'information' sheet on the exam will be distributed. That information sheet will cover the format and content of the exam.

Failure to notify the instructor that the Midterm exam will be missed will result in a makeup that *might* be harder, *not by design, but by consequence of being different*. Missing the Final Exam will result in a 0 on the Final Exam so that grades can be submitted on time before fleeing the state.

The Midterm Exam will be one hour long; the Final Exam will be two hours long. Both exams will be paper-based, closed note /closed book / closed computer, and cumulative up to that point.

Note: In order to pass the course, you will need an overall average of 60 in the course and you will need a 60 or above on the Final Exam.

Cheating Policy

Don't cheat!

Cheating is not allowed on Homework, Programming Labs, Programming Projects, and Exams. All parties involved in cheating will receive a 0 and will be reported to the dean. Excessive

cheating within the class can result in an F for the course. Excessive cheating at RVCC can result in expulsion. Consult your student manual.

For the Programming Labs, asking the instructor for hints is not considered cheating. It is ok to email a Java file to the instructor to ask for assistance. But, don't expect a response after 10:15 pm on any night of the week/weekend.

NOTE: you have permission to use the instructor's demo .java files, in whole or in part, for your programming labs and programming projects (just cite that in a comment in the code) and you have permission to use the code the instructor writes on the whiteboard (just cite that in a comment in the code) and you have permission to use the code in the book, in whole or in part (just cite that in a comment in the code).

Late Policy

Don't submit work late!

According to the RVCC Catalog, students are not to be penalized for 1 week of absences. To accommodate this, at the end of the semester the lowest homework, the lowest programming lab and the lowest Data Structures lab will be dropped.

"Life Happens"

According to the RVCC Catalog, students are not to be penalized for 1 week of absences. To accommodate this, at the end of the semester the lowest Homework, the lowest Programming Lab and the lowest Data Structures lab will be dropped. That handles "life happening".

Extra Credit

Some exams/homework/labs/projects contain extra credit questions/opportunities. Other than that, no extra credit opportunities will be provided.

For example, if you choose not to submit ANY homework and then in April you ask for "Extra Credit" to make up for the missed homework, the answer is NO.

Additional Policies:

Classroom Behavior

Cell Phones:

For the "lectures" please silence cell phones. It is understood that there may be times when emergency calls occur or the "pick me up at the airport" call needs to happen. For those calls please step into the hallway to take the call. During lab time, you can turn the sounds back on the cell phones.

Talking:

For the "lectures" please don't talk. It is understood that you may need to turn to a neighbor for the occasional question like "what slide is he on?" or "which file is that?" or "what page # did he say?" and that's fine. It is the full conversations that are distracting not only to the instructor but to the class as a whole.

For the Lab Time you can talk all you want, but "indoor voices" please.

Language:

The instructor will try his darn-doodliest to not swear during class time but an occasional expletive may slip out. Please try your darn-doodliest not to swear but don't fear reprisals if an occasional expletive slips out.

Proper Use of Computers:

Lecture: the instructor is easily distracted by typing and would prefer that computers are not used during the lecture, except for opening and view demo Assembly files. Pointing and laughing at the monitors is very distracting. ***Printing during lecture is extremely distracting and very upsetting.*** If class members repeatedly distract the instructor, then all computers and laptops must be closed down for the rest of the semester.

Lab: During lab time everyone should be working on the labs/projects in Assembly. None of the labs/projects require social media sites like Facebook.

Bathroom Breaks:

Lecture: Depending on the length of the lecture there may be a break partway through, and usually it will be announced at the start. If you need to leave for a break just quietly get up and leave.

Lab: take all the breaks you need.

Class Attendance:

See "RVCC Student Resources" for official policy.

Note: sometimes attendance is taken based on what graded items were not returned. If you come in late and see that graded items were returned, please wait for a break or Lab Time to get your graded items

Student Handbook

You are responsible for all policies stated in the Student Handbook, including Academic Integrity Policy and Code of Student Conduct.

In Canvas, there is a tab on the left "RVCC Student Resources", it is there

Withdrawal Procedure

See school's web page for Spring 2024 Withdrawal and Refund Schedule and Refund Info (see:

<https://commons.raritanval.edu/admin/finance/Documents/Spring%202024%20Withdrawal%20and%20Refund%20Schedule.pdf>)

(see: https://commons.raritanval.edu/admin/finance/Pages/refund_info.aspx)

Class Schedule

Please see the Class Schedule for the listing of lecture topics and timing of homework / labs / projects / exams

Syllabus Part 2 - College Policies and Resources

In Canvas, there is a tab on the left "RVCC Student Resources", additional policies and information about resources are found there

:)