**Montgomery College  
Department of Engineering, Physical, and Computer Sciences  
Rockville, MD**  
***CMSC 204 Computer Science II, CRN 32757***

***Spring 2020***

Course Syllabus

|  |  |
| --- | --- |
| **Professor:** | Farnaz Eivazi |
| **Course Hours:** | 6:30 pm - 9:50 pm Wednesdays |
| **Course Location:** | High Tech and Science Center 203 |
| **Office:** | NA |
| **Office Hour(s):** | By appointments only |
| **Email:** | farnaz.eivazi@montgomerycollege.edu |
| **Course Site:** | bb-montgomerycollege.blackboard.com |
| **Final Exam:** | May 13th 7:15-9:15pm |

**Course Description**

**Course Title**: Computer Science II **Credit Hours**: 4

**Montgomery College Catalog Course Description:**

Continues ideas introduced in CMSC203, emphasizing writing larger programs and designing and implementing classical abstract data types such as list, stack, queue, binary search tree, graph, priority queue, hash table. Topics include string processing and recursion; data abstraction, encapsulation, and structure implementation; object-oriented program design; specification, implementation and application of these traditional ADTs. The course also emphasizes dynamic memory allocation, search and sorting algorithms, and introduces algorithm complexity. Designing and implementing advanced level programming assignments are an integral part of the course.

**PREREQUISITE**:

Assessment Levels/Prerequisites: A grade of C or better in CMSC203.

Pre- or corequisite: MATH182

**Delivery**: Thisis a face-to-face lecture course. All assignments, labs and exams will be administered through Blackboard**.**

**Outcomes**Upon completion of this course, students should be able to:

* Apply the principles of networking, exceptions, and error-handling in a Java environment
* Contrast basic concepts of procedural and object–oriented programming
* Demonstrate basic principles of program development and design
* Demonstrate the implementation of abstract data types, such as list, stack, queue, priority queue, binary search tree, graph, and heap
* Describe the design and time complexity of algorithms.
* Utilize fundamental features of a higher level language, including event-driven programming, graphical user interface, and multi-threading.

***Content Outline***

* Software Engineering Principles (Design and Verification)
* Data Design and Implementation (Built-In, Abstract Data Types, Classes)
* Encapsulation, Inheritance and Polymorphism
* Collections
* List ADT (Sorted and Unsorted)
* Vectors
* Stack and Queue ADTs
* Linked Structures
* Recursion
* Binary Search Tree ADT
* Heap and Priority Queue ADTs
* Graph ADTs
* Sorting and Searching Algorithms
* Java networking using sockets
* Multithreading

**Textbooks**

Data Structures and Abstractions with Java, 5th Edition Frank M. Carrano and Timothy M. Henry (ISBN 9780134831695/0134831691)

Alternate: Data Structures and Algorithms Using Java, William McAllister, Jones and Bartlett (ISBN 978-0-7637-5796-4)

**Software & Supplies**

**Java IDE and SDK**

**You will need a Java IDE (Integrated Development Environment) with a Java SDK (Software Development Kit). I suggest Eclipse, because that is what is available in Montgomery College labs and it can be downloaded for free at www.eclipse.org. The Javadoc and JUnit test tools are included with Eclipse. If you choose to use an IDE other than Eclipse, you will still be required to turn in Javadoc and JUnit tests. I will be using Eclipse to grade your programming assignments. If the program doesn’t run correctly using Eclipse, the assignment grade will suffer due to inability to run and test.**

**Safari books online - ProQuest**

**We may be using parts of various textbooks available through Safari books online - ProQuest. Access to this database is free through myMC->Libraries.**

**A Web browser**

**Internet Explorer, Firefox, Chrome, etc.**

**Adobe Acrobat fileviewer**

**This can be downloaded for free at** [www.adobe.com/products/acrobat/readstep2.html](http://www.adobe.com/products/acrobat/readstep2.html)

**Powerpoint fileviewer**

**This can be downloaded for free at**

[www.microsoft.com/en-us/download/details.aspx?id=13](http://www.microsoft.com/en-us/download/details.aspx?id=13)

**Computer Lab Usage/Privileges**  
Computer labs are available for course related work.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Course Location*** | ***Lab Location*** | ***Overflow Lab*** | ***Office Location/Phone*** |
| HU building | H314 | H312 | H307/ 567-5187 |
| CS building | CS25/26 & CS21A | H314 | CS19 / 567-5156 |

Labs are open Monday through Friday 8:30 am - 10:00 pm, Saturday 8:30 am - 4:00 pm, and Sunday 8:30 am - 2:00 pm. Labs open on the first day of the semester and close on the last day of finals and all College holidays. Any changes in hours will be posted. Rules and guidelines governing the use of labs are published by the Computer Lab Offices.

**CS Tutoring**

There is a tutoring lab for CMSC/CMAP in the Ackerman Learning Center. Please review the posted schedule to see when tutors are available to help you with your course.

**Grading**

Course grades will be based upon the following:

* There will a midterm and a final exam.
* There will be seven programming assignments/projects.
* There will be several graded labs. The lowest grade will be dropped.
* All exams will be taken online, all programming assignments and labs will be submitted online.

The relative weights of these assignments are:

|  |  |  |
| --- | --- | --- |
|  | ***% of Course Grade*** | ***Approximate Date*** |
| Midterm I Exam | 15 | Feb 26th |
| Midterm II Exam | 15 | April 20th |
| Assignments | 25 | Bi-weekly |
| Labs | 15 | weekly |
| Final Exam | 30 | May 13th 7:15-9:15pm |

A=100-90%   B= 89-80%   C=79-70%   D=69-60% F=60%-below

**Late & Make-up Policies**

**Late Assignments/Labs**

Assignment designs are generally due one week after assignment, and implementation due two weeks after they are assigned, as specified by the professor. Labs are generally due one week after they are assigned. No late assignments or labs will be accepted.

**Make-Up Exams**

There are no make-up exams.

**Final Exam**

The final exam is taken in class according to the final exam schedule.

**Academic Honesty**   
Academic dishonesty in college is a very serious offense. Each student is expected to do his/her own workon all quizzes and tests and class and homework exercises. Students who engage in any act that the classroom instructor judges to be academic dishonesty or misconduct are subject to sanctions. For more information, please refer to Number 42001 in [**http://cms.montgomerycollege.edu/pnp/#Chapter\_4**](http://cms.montgomerycollege.edu/pnp/#Chapter_4)

**Academic Honesty in CMSC 204**

All students are expected to do their own work. You may receive insights, editing and debugging help from tutors, fellow students, and acquaintances, but ***you cannot share code.***  **You are expected to do all homework assignments by yourself – DO NOT GIVE YOUR CODE TO YOUR FRIENDS!** You are responsible for completing the assignments yourself. (Your fingers are the only ones that should touch the keyboard). If your assignment uses features of the language that have not yet been covered in this course, you may be asked to defend your work. Academic dishonesty will **not** be tolerated. Programming assignments will be submitted to a plagiarism-checking utility which compares each submission with others, and rates them by similarity. If you turn in the same assignment or uncommonly similar assignment to another student (past or present), or if your assignment is uncommonly similar to code found on the internet, you will receive 0 points for the assignment, and potentially a grade of “F” for the course. You will be reported to the Dean of Student Development.

**UMD CMSC 203 and CMSC 204 course transferability:**

**CMSC 203 is only transferrable to UMD** if completed Fall 2018 or Later.

**CMSC204 is only transferable to UMD if:**

CMSC203 was completed Fall 2018 or later **AND** CMSC204 was completed Spring 2019 or later OR

AP Computer Science A exam score of 5 (which is equivalent to CMSC131) **AND** CMSC204 was completed Spring 2019 or later.

If a student took CMSC203 at MC prior to Fall 2018 OR transferred a course to MC that is equivalent to CMSC203, they will need to do one of the following:  
1. Retake the current CMSC203 course at MC before enrolling in CMSC204 at MC  
2. Plan to take CMSC131 at UMD in their first semester.\*

\*Students who fall outside of these course-taking patterns will not be eligible to take the CMSC131 exemption exam at UMD and will be required to enroll in CMSC131 in their first semester, and therefore would not be eligible for the CS major until they complete the [Gateway requirements](https://lep.umd.edu/cs-lep.pdf): CMSC131, CMSC132, MATH140.

**UMBC CMSC 203 and CMSC 204 courses’ transferability:**

CMSC 203 and CMSC 204 are transferrable to UMBC if completed with grade B or above

**Communication**

**College Email**

The Montgomery College e-mail account is the official means of communication. Check your email account routinely for course announcements, invoices, important admission/registration information, waitlist status etc. I do not regularly check Blackboard messages. The best way to contact me is via email at  [farnaz.eivazi@montgomerycollege.edu](mailto:%20farnaz.eivazi@montgomerycollege.edu).

**General Course Information and Policies**

Thisis a face-to-face lecture course. However, all assignments, discussions and Midterm I will be administered through Blackboard. Midterm II and the Final Exam will be paper exams. Completion of all assignments is necessary in order to pass this course.

**A. Technical Requirements & Technical Support Policy**

You will be uploading all assignments/labs to the Blackboard website.

You will need the following to participate in this class:

* + Regular use of a computer with Internet access. You may use computers in the computer labs in Humanities and Computer Science (see “Computer Lab Usage/Privileges” above for locations and hours). Expect to spend several hours online each week.
  + A web browser such as Internet Explorer or Firefox.
  + Java IDE and SDK (available at [www.eclipse.org](http://www.eclipse.org))
  + A file viewer for [Adobe Acrobat](http://www.adobe.com/products/acrobat/readstep2.html) and powerpoint files.

*For technical assistance*with college supported resources, call the Montgomery College IT Service Desk at 240-567-7222.

*The HELP link on the left-hand course menu*links to the MC Blackboard Online Support Center. Students can

1. Call the Support Center at 240-567-7222, or

2. Chat with a service representative, or

3. Submit a ticket.

The *My Support* link at the top of the Blackboard Online Support Center screen links to a history of one’s correspondence with the support center.

*System Downtime*

The Office of Information Technology conducts computer network maintenance on Sunday morning from 12:01 AM to 6:00 AM each week. During this time you may be not be able to access My MC to login to Blackboard. Do not rely on this time to submit course work.

You are responsible for the operation of the computing system you use off campus. A malfunctioning computer system is not a valid excuse for submitting late work. ***Back up your work.***

If technical difficulties with university computers or network servers affect assignments, quizzes, exams, or scheduled class presentations, your instructor will use the following procedures to communicate with you to provide appropriate directions in the order listed below:

• An announcement will be posted on the course site.

• If the course site is unavailable, an e-mail will be sent to all students through the Montgomery College email.

**B. Email Policy**

Excessive or unnecessary emails make considerable time demands on both the sender and the recipient. Before sending an email, be self-sufficient: check the course website (including syllabus, assignment descriptions, announcements, etc.) to find the answer to your question.

I check my MC email at least once a day Monday through Thursday and at least once during the weekend (Friday through Sunday). Please plan accordingly.

Do not email me asking to debug code. I will not debug your code for you. The debugging process is so vital to being a successful programmer that you must learn this process. HoHIf you are in need of debugging help, please visit me during office hours, or we can arrange a time to meet. At that time I can ask questions about your code, review your design and help point you in the right direction.

Email Policy Guidelines:

* Use my Montgomery College e-mail (farnaz.eivazi***@montgomerycollege.edu***) for all email. I do not check messages on Blackboard regularly.
* Include “**CMSC 204**” as part of the **SUBJECT** line, to alert me that the email refers to this course.
* Generally, I will respond to e-mails within 24 hours during the week and 48 hours over the weekend or holidays.

**C. Assignment Policy**

* All assignments/labs are submitted electronically on the Blackboard website, before or on the day and time that it is due. All files needed for an assignment/lab/worksheet must be submitted together in a .zip file. If one file is submitted late, the entire assignment will be considered late. You will be submitting source code and junit tests (.java), javadoc (.html zipped with the doc directory), design files (.docx, .pdf, or images), UML diagram files (images) and screenshots (images, .docx or .pdf).
* If there are technical problems with the course site, assignments can be sent to my Montgomery College e-mail (farnaz.eivazi@montgomerycollege.edu). The subject line of the e-mail must include the course name (CMSC 204) so I will be alerted that it is in reference to the course.
* Format for assignments: Submit two compressed files, named with your last name/first name and the assignment number, following the below example. One compressed file will contain all the files required for submission, and the other will only contain the .java files, for submission to the plagiarism-checking utility. Use this example:
  + LastnameFirstname\_AssignmentX.zip
    - doc *(directory)*
      * Javadoc html files
      * other infrastructure directories
    - src *(directory)*
      * File1.java
      * File2.java
      * File1Test.java *(provided by instructor, w/ STUDENT tests)*
      * File2\_Interface.java *(as provided by instructor)*
    - UML diagram.jpg *(or docx or other format)*
    - Screenshot1.jpg *(if required)*
    - Screenshot2.jpg *(if required)*
  + LastnameFirstname \_AssignmentX\_JavaFiles.zip
    - File1.java
    - File2.java
    - File1Test.java
    - File2\_Interface.java
* If you need to discuss your grade or feedback you received from your instructor on an assignment, make an appointment with your instructor. This may be done in a visit during published office hours or via e-mail or telephone contact.
* Students must ensure that assignment/lab files are free of viruses before submitting them. Keep your virus detection software up to date. Should an assignment file fail scrutiny by our institution’s standard virus detection software, the student submitting it will be so advised by e-mail. Any subsequent failure to adhere to this requirement will cause an assignment to be unacceptable.
* You have the ability to submit the assignment/lab/worksheet multiple times, until it is due. I will only grade the most current submittal and it must contain all the files needed for the assignment/lab/worksheet. In other words, if you submit your assignment and then discover that you forgot to include your screenshots, or need to make a change in one of your source code files, you must submit all needed files in your final submittal, not just the forgotten file or the changed file.
* ***Backing up your work***

In this course you will complete most of your assignments and labs on a computer. You are responsible for ensuring the safety of your work by making regular backups (extra copies). “The computer ate my homework, I lost my flash drive, my hard drive crashed, or my printer isn’t working,” are **not acceptable excuses**. Make frequent backups of your work and save the work in multiple places. When you work in the classroom or in the lab, be sure to save your work on a device OTHER THAN the computer you are working on, since they are automatically wiped at midnight daily. You can retrieve files that were successfully uploaded to the Blackboard server, but it is always a good idea to keep backup copies of all your work. I would recommend using Dropbox ([www.dropbox.com](http://www.dropbox.com)) or Git ([www.GitHub.com](http://www.GitHub.com)) or something similar to keep a copy of your files externally. Students are responsible for keeping a copy of all graded assignments. If there is no copy of graded work in question, no grade change or credit for a missing assignment is possible.

**D. Self-motivation and Self-direction policy**

Expectations of students with respect to self-motivation and self-direction in an e-learning environment.

* Be self-motivated and self-disciplined.
* Be a good time manager.
* Approach the course with a desire to learn.
* Assume a leadership role, and be a teacher when necessary. Voluntarily help other students, bearing in mind, however, that doing other people’s work for them is tantamount to cheating.
* Develop needed technology skills.
* Contribute to course discussions. Listen to others, and respond respectfully to their comments.
* Contribute to team activities, and respect the ideas of others.
* Comply with all course policies.
* Submit constructive suggestions for improving the course.
* **This course is hard – you will need to allocate adequate time to study and programming.**

**The Ackerman Learning Center** is located on the [Rockville Campus](http://www.montgomerycollege.edu/about-mc/campuses-and-locations/rockville-campus/index.html) on the ground floor of the new Science West (SW) building in room 109. Contact them at 240-567-5200

**Collegewide Policies**

**Important Student Information Link**

In addition to course requirements and objectives that are in this syllabus, Montgomery College has information on its web site (see link below) to assist you in having a successful experience both inside and outside of the classroom. It is important that you read and understand this information. The link below provides information and other resources to areas that pertain to the following: student behavior (student code of conduct), student e-mail, the tobacco free policy, withdraw and refund dates, disability support services, veteran services, how to access information on delayed openings and closings, how to register for the Montgomery College alert System, and finally, how closings and delays can impact your classes. If you have any questions please bring them to your professor. As rules and regulations change they will be updated and you will be able to access them through the link. If any student would like a written copy of these policies and procedures, the professor would be happy to provide them. By registering for this class and staying in this class, you are indicating that you acknowledge and accept these policies.

[http://cms.montgomerycollege.edu/mcsyllabus/](https://mail.montgomerycollege.edu/owa/redir.aspx?C=pYgo18BD9EO-gncv_ksLttz-gtFZstJIwKJd3G8hgrj60F3WroTNafobL208ZOzbUsAEJavcdxk.&URL=http%3a%2f%2fcms.montgomerycollege.edu%2fmcsyllabus%2f)

**Basic Needs Statement**

“Any student who has difficulty accessing sufficient food to eat every day, or who lacks a safe and stable place to live, is urged to contact the *Dean of Students Affairs* on your campus. Furthermore, please notify the professor if you are comfortable in doing so. This will enable the professor to provide any resources that they may possess.  We know this can affect performance in the course and Montgomery College is committed to your success.”

**The Deans of Student Affairs**

Germantown: Dr. Jamin Bartolomeo, [jamin.bartolomeo@montgomerycollege.edu](mailto:jamin.bartolomeo@montgomerycollege.edu)

Rockville: Dr. Tonya R. Mason (RV), [tonya.mason@montgomerycollege.edu](mailto:tonya.mason@montgomerycollege.edu)

Takoma Park/ Silver Spring: Dr. Clemmie Solomon, [clemmie.solomon@montgomerycollege.edu](mailto:clemmie.solomon@montgomerycollege.edu)

**STUDENT *HEALTH AND WELLNESS/ FUEL FOR SUCCESS* WEBSITE**

This website offers information about resources for food on our campuses and in the community and has links for community resources. The site offers the schedule for the mobile markets, locations of the food pantries as well as a link for those who wish to contribute their time or money to support our students <http://cms.montgomerycollege.edu/student-health-and-wellness/fuel-for-success/>

**CAMPUS FOOD PANTRIES**

Each of the main campuses has a pantry stocked with snacks and food. Students are welcome to come pick up a snack to carry you through your next class, and to take a few items home.

|  |  |  |
| --- | --- | --- |
| **Campus** | **Pantry Location** | **Days & Hours of Operation** |
| **Germantown** | *High Tech (HT) Food Pantry* In the hallway, near HT300 | 9:00 a.m. - 5:00 p.m. |
| **Rockville** | *Women’s’ and Gender Studies Program*  *Food Pantry*  Hallway outside of MT212  ---  *Biology Department Food Pantry*  Science Center, 2nd floor hallway | Monday-Friday  9:00 a.m. - 5:00 p.m.  ---  Monday-Friday  9:00 a.m. - 5:00 p.m. |
| **Takoma Park/Silver Spring** | *Commons Food Pantry*  CM 110  ---  *Institute for Justice, Race and Civic Engagement Food Pantry*  Pavilion 4, #203  [Vincent.intondi@montgomerycollege.edu](mailto:Vincent.intondi@montgomerycollege.edu) | Monday - Thursday  8:00 a.m. - 7:00 p.m.  Saturdays 9:00 a.m. - 1:00 p.m.  ---  Monday – Friday |

**MOBILE MARKETS**

The College has a partnership with the Capital Area Food Bank. They distribute a variety of foods at Mobile Markets, at each of the three main campuses. Food is available on a first-come basis. No proof of eligibility is required. The markets are open to the entire community. For information on volunteering at the Mobile Markets, reach out to Student Affairs, [Carmen Poston-Travis](mailto:carmen.poston@montgomerycollege.edu) 240-567-5253 or [Benita Rashaw](mailto:benita.rashaw@montgomercollege.edu) 240-567-4389.

|  |  |  |  |
| --- | --- | --- | --- |
| **Campus** | **Mobile Market Location** | **Time** | **Dates** |
| **Germantown** | Outside SA (In inclement weather - High Tech Building (HT),   Second Floor Upper Lobby) | 10:00 a.m.- 12:00 p.m. | September 12, 2018 October 10, 2018 November 7, 2018 December 5, 2018 February 13, 2019 March 20, 2019 April 10, 2019 May 8, 2019 |
| **Rockville** | Outside of the North Garage (NG) | 10:00 a.m. - 12:00 p.m. | September 19, 2018 October 17, 2018 November 14, 2018 December 12, 2018 February 20, 2019 April 17, 2019 May 15, 2019 |
| **Takoma Park/Silver Spring** | Outdoor space  between the Student Center and North Pavilion  (In inclement weather - ST atrium) | 2:00 p.m. - 4:00 p.m. | September 5, 2018 October 3, 2018 October 31, 2018 November 28, 2018 February 6, 2019 March 6, 2019 April 3, 2019 May 1, 2019 |

**SCHEDULE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Module** | **Labs/Study Guides** | **Assignments** |
| **Week 1** | **Introduction Module**  **Review Module**  **Module 1- ADT Bag**  **Module 2 - Exceptions**  **Review Module** | *GitHub lab is assigned*  NClass Video  Eclipse Video  Exceptions Video  Study Guide - Exceptions | ***Assignment 1 (Passwords)***  Classes – Data Element (Abstract Base and Inherited classes), Data Manager, Exceptions and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element (Base and Inherited classes), Data Manager, Exception, GUI driver classes  Javadoc  Junit tests (student created)    Features:  User created exceptions, read/write files (FileChooser), inheritance/polymorphism |
| **Week 2** | **Module 3 - Generics**  **Module 4 – Array based ADT**  **Module 5 – Linked ADT**  **Module 6 – Efficiency of Algorithms** | *GitHub Lab is due*  *Generic Lab is assigned*  Study Guide –  Generics  Generics Video  Study Guide – Lists  Study Guide – Big-O  Notation | ***Assignment 1 Design is due*** |
| **Week 3** | **Module 7 – Stacks**  **Java Interlude 7 – Inheritance and Polymorphism**  **Module 8 – Queues** | *Generic Lab is due*  *JUnit Lab is assigned*  Study Guide – Restricted structures | ***Assign 1 implementation is due***  ***Assignment 2 (Notation)***  Classes – Data Element, Data Structure (Generic Queue, Generic Stack), Data Manager, Exception and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element, Data Structure (Generic Queue, Generic Stack, Data Manager, Exception, GUI driver classes  Javadoc  Junit tests (student created)  UML  Features:  Generic Queue, Generic Stack, exception handling, read/write files (FileChooser) |
| **Week 4** | **Module 9 – Recursion**  **Java Interlude 6 – Mutable and Immutable Objects**  **Module 10 – Lists** | *JUnit Lab is due*  *Recursion Lab is assigned*  Study Guide –  Recursion  Recursion Video | ***Assignment 2 design is due*** |
|  | **MIDTERM I EXAM** (90 min) |  |  |
| **Week 5** | **Module 11 – Iterators**  **Module 12 – Dictionaries** | Recursion Lab is due  Study Guide – Iterators and  Interfaces  Iterator Video | ***Assignment 2 implementation is due***  ***Assignment 3 (Linked Lists)***  Classes – Data Element, Data Structure (Generic Linked List, Generic Sorted Linked List), Data Manager, Exception and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element, Data Structure (Generic Linked Lists, Generic Sorted Linked List), Data Manager, Exceptions, GUI driver classes  Javadoc  Junit tests (student created)  UML  Features:  Generic Linked List with iterator, Generic Sorted Linked List using a Comparator (inherits from Generic Linked), exception handling |
| **Week 6** | **Module 13 – Searching and Hashing** | Hashing Lab is assigned | ***Assignment 3 design is due*** |
| **Week 7** | **Module 14 – Trees** | Hashing Lab is due  Tree Lab is assigned  Study Guide - Trees | ***Assignment 3 implementation is due***  ***Assignment 4 (Concordance)***  Classes – Data Element, Data Structure (Hash Table), Data Manager, Exception and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element, Data Structure (Hash Table), Data Manager, Exception, GUI driver classes  Javadoc  Junit tests (student created)  UML  Features:  Hash Table, hash code, buckets/chaining, exception handling, read/write files (FileChooser) |
|  | **Spring Break – Mar 11-15** |  |  |
| **Week 8** | **Module 15 – Cloning**  **Module 16 - Sets and Maps**  **Module 17 – Advanced I/O and Object Streams** | Study Guide – Sets and Maps  Serialization Video | ***Assignment 4 design is due*** |
| **Week 9** | **Module 18 – Sorting**  **Module 19 – Heaps** | Heap Sort Lab is assigned  Study Guide - Sorting | ***Assignment 4 implementation is due***  ***Assignment 5 (Morse Code)***  Classes – Data Element, Data Structure (Binary Search Tree), Data Manager, Exception and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element, Data Structure (Binary Search Tree), Data Manager, Exception, GUI driver classes  Javadoc  Junit tests (student created)  UML  Features:  Binary Search Tree with Comparator, exception handling, read/write files (FileChooser) and Read/write trees with object streams |
|  | **MIDTERM II EXAM** (90 min) |  |  |
| **Week 10**  **Week 11** | **Module 20 - Graphs** | Heap Sort Lab is due  Graph Lab is assigned  Graph Lab is due  Study Guide - Graphs | ***Assignment 5 design is due***  ***Assignment 5 implementation is due***  ***Assignment 6 (Town Graph)***  Classes – Data Element, Data Structure (Graph), Data Manager, Exception and GUI driver  Provided: GUI template, Interface for Data Manager, Junit tests  Student: Data Element, Data Structure (Graph), Data Manager, Exception, GUI driver classes  Javadoc  Junit tests (student created)  UML  Features:  Graph, exception handling, read/write files (FileChooser) |
| **Week 12** | **Module 21 – Concurrency, Multithreading** | Thread Lab is assigned  Study Guide – Threads | ***Assignment 6 design is due*** |
| **Week 13** | **Module 22 – Internet Networking** | Thread Lab is due  Chat Room Lab is assigned  Study Guide – Internet Networking | ***Assignment 6 implementation is due*** |
| **Week 14** | **Module 17 – Other Data Structures** | Huffman Tree Lab is assigned  Sockets Lab is due  Chat Room Lab is due  Review |  |
|  | **FINAL EXAM** (120 min) | Wed Dec 18, 7:15-9:15pm  Huffman Tree Lab is due |  |