General Information
CSC 207, “Algorithms and Object-Oriented Design”
Department of Computer Science
Grinnell College

In CSC 207, we’ll study an object-oriented model of computation for problem solving using practical organization of data and algorithms that operate on these structures. To implement this computation model, we’ll use a tool that provides an environment for developing programs with Java as an object-oriented programming language. An important feature of the object-oriented computation model is to separate implementation details of data structures/algorithms from their usages in the programs.

After completing this course, students gain the following learning outcomes:

- Understanding
  1. Difference between primitive versus reference data types.
  2. Visibility access of class members from outside world.
  3. Difference between abstract classes, concrete classes, and interfaces.
  4. Difference between string and string builders.
  5. Basic commands of Git as a source code versioning system.
  6. Difference between arrays and array lists.
  7. Equivalence relations and disjoint sets.

- Creating
  8. Objects for encapsulating fields and methods that access or manipulate fields.
  9. Generic methods that process an input parameter with multiple data types.
  10. Classes that inherit fields and methods from existing classes and define new fields or methods.
  11. Streams for reading from or writing into files and terminals.
  12. Exception objects that provide information about the cause and location of errors. In addition, students create code blocks that catch and handle the exceptions.
  13. Random number generators for integer and double data types.
  15. Minimum spanning tree of graphs by implementing the Kruskal’s algorithm.
  17. Comments in Java source files using the Javadoc tool provided by Eclipse as an Integrated Development Environment.

- Testing
  18. The order of elements sorted by multiple sorting algorithms using comparators.
  19. Program correctness by using JUnit unit testing framework and debugger provided by Eclipse.

- Searching
  20. Standard java library for using appropriate built-in classes bundled into packages.

- Analyzing
  22. The loop structure of algorithms to estimate their asymptotic running times using Big-O notation.
  23. The heapsort algorithm by building binary heaps from unordered data.

- Applying
  24. Classical data structures (e.g., lists, stack, queues, hash map, set) in Java Collections API to solve problems.
  25. The following algorithms on graphs for finding the shortest paths:
     a. Breadth-first search
     b. Dijkstra’s algorithm
c. Topological sorting

26. The following algorithms for balancing binary search trees:
   a. AVL trees
   b. Red-Black trees

27. Iterators to access each element of underlying data structures in constant time.

The class meets online using Blackboard Collaborate on Mondays, Tuesdays, Wednesdays, Thursdays, and Fridays from 7 pm to 8:30 pm Central Standard Time beginning on Thursday, April 1.

The following is the link to the course Web page:

   http://hajiamini.cs.grinnell.edu/courses/CSC207/

The lecturers will be pre-recorded and placed in Pioneer Web before the class meeting time.

Resources


Professor Weiss’s Web page: https://users.cs.fiu.edu/~weiss/
Data Structures and Problem Solving Using Java source code:
Data Structures and Problem Solving Using Java errata:
https://users.cs.fiu.edu/~weiss/dsj4/errata/

As a secondary resource, you can read about the basics of Java from the following URL. The contents of this URL are written by ORACLE Corporation who owns the Java programming language:

https://docs.oracle.com/javase/tutorial/

Please refer to topics that are discussed under Trails Covering the Basics in the above URL.


The Java Language Specification and Java Standard Library:
https://docs.oracle.com/javase/specs/jls/se11/jls11.pdf
https://docs.oracle.com/en/java/javase/11/docs/api/allclasses.html
Open Java Development Kit (Version 11): http://openjdk.java.net/
Eclipse project page: https://www.eclipse.org/

For remote pair programming, you can use the following Eclipse plug-in. This plug-in facilitates collaborative live coding and communication through a chat box. With this plug-in, you can do live

https://marketplace.eclipse.org/content/saros-distributed-collaborative-editing-and-pair-programming

After installing the plug-in, you can see Saros on the menu bar of Eclipse desktop IDE. From Saros, choose Getting Started, which gives a tutorial for setting up Saros on your Eclipse IDE.

Note: Please do not install or use Eclipse and Saros before April 14 when we learn about these tools.
The Instructor and Mentors

The instructor for this course is Shervin Hajiamini. My e-mail is hajiamini@grinnell.edu.

My office hours for spring 2021 are Monday through Friday, 8:30 pm to 9 pm. If you need to meet me outside these office hours, please email me your suggested times for setting up an appointment. I’ll choose a time that works for both of us.

The mentor for lab exercises is Robert Lorch (lorchrob@grinnell.edu). The mentor will hold a mentor session weekly. Take advantage of these sessions for obtaining the better understanding of concepts that are taught in the class. You may also use the mentor sessions to prepare for quizzes and discuss ideas for solving the programming assignments.

Reading Assignments

You are expected to do specified reading assignments for each class session. Questions for each reading assignment are related to a topic that is scheduled in a given day. Submit your answers to those questions before 7 pm when we meet in the class. You receive points only if you submit your answers by the deadline. Your answers will not be graded, but doing those carries some portion of your final grades.

Exercises

There are two types of exercises: 1) in-class labs and 2) Programming Assignments (PAs). The labs and PAs will be posted on the course web page. The schedule of labs and PAs is in Schedule of Topics document on the course webpage.

You’ll be working in pairs on the labs for most of the class sessions. We’ll be using the breakout sessions of Blackboard Collaborate for screen sharing, audio and video if bandwidth permits. You and your partner will do the lab works using the driver/navigator pair programming method. During the lab work, you should switch roles, every 15 to 20 minutes. If you won’t complete the labs during the class time, you need to arrange times to meet outside of class. After finishing a lab, make sure to keep a copy of the lab source files somewhere on your MathLAN home directory. For all labs given in a week, it is required that you submit your works by 11:59 pm on Saturday of that week. If you submit a lab work late, you lose point for that lab. Note that only one submission is required for each lab. The names of lab partners must be on the submitted source files.

You are required to submit your solutions for PAs. The submission time of each PA will be 11:59 pm on a date given in the description of the assignment. I will give you some startup codes for the assignments. For each day you submit an assignment late, a penalty of 10% is applied to your assignment grade.

Besides Java source files, your lab and PA submissions should also have a readme text file that specifies command lines for compiling and running your programs. If a lab or PA requires generating specific outputs from programs, you need to include them in your readme file. Please upload your submissions to designated entries on Blackboard.

Class Attendance

Class attendance is mandatory. For any unexpected absence, please email me and your class mentor at least a day before the date that you will be absent.

If you miss a class, you need to complete and submit the lab assigned for that day by its deadline.

The Exams

We will have weekly quizzes that will be given on Tuesdays. (The schedule of quizzes is in Schedule of Topics.) Deadlines for submitting quizzes are on Fridays, 11:59 pm. When taking a quiz in a given week, the quiz questions will cover topics that are discussed in the preceding week. Quizzes have time limits (typically between 30 minutes to 45 minutes), but you can take them any time by the Friday of weeks when quizzes will be given.
The midterm exam and final exam will be take-home exams. The date and time of the exams are in Schedule of Topics. Exam questions will include concepts covered in the class meetings. They will also include questions about code snippets.

Tutoring
The Computer Science Department provides tutors for CSC 207. The tutoring times are:

- Sundays, 3 pm – 5 pm and 7 pm – 10 pm
- Mondays to Thursdays, 4 pm – 7 pm

You can meet tutors on the CS Team using a channel called Drop in Tutoring.

Peer tutors may also be available for regular, more intensive one-on-one tutoring. As the course gets underway, please let me know if you’re interested in regular weekly individual tutoring.

Grading
The final grade will be based on the reading assignments (5%), PAs (45%), labs (10%), quizzes (10%), midterm (15%), and the final exam (15%)

The following grading scale determines your final grade:

<table>
<thead>
<tr>
<th>Percentage grade</th>
<th>Letter grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>93-100%</td>
<td>A</td>
</tr>
<tr>
<td>90-92%</td>
<td>A-</td>
</tr>
<tr>
<td>87-89%</td>
<td>B+</td>
</tr>
<tr>
<td>83-86%</td>
<td>B</td>
</tr>
<tr>
<td>80-82%</td>
<td>B-</td>
</tr>
<tr>
<td>77-79%</td>
<td>C+</td>
</tr>
<tr>
<td>70-76%</td>
<td>C</td>
</tr>
<tr>
<td>60-69%</td>
<td>D</td>
</tr>
<tr>
<td>0-59%</td>
<td>F</td>
</tr>
</tbody>
</table>

Copyrights and Licensing
The Grinnell College copyright policy is stated in the following link:

https://www.grinnell.edu/sites/default/files/documents/copyright_0.pdf

Based on this policy, a student or a group of students who write(s) source codes or texts for this course retain(s) copyright to them. I encourage you to use the GNU General Public License (Version 3) for source codes and the Creative Commons Attribution-ShareAlike International license (Version 4) for text:

https://www.gnu.org/licenses/gpl-3.0.en.html

https://creativecommons.org/licenses/by-sa/4.0/deed.en

These licenses state that your codes and texts can be copied, modified, and redistributed freely only if they are released with the same license.

Note that the College has the right to distribute students’ works within the College community for instructional or administrative purposes without paying any royalty to the students.
Collaboration and Plagiarism
I will evaluate your academic performance individually. Therefore, it is not ethical to borrow other people’s works without acknowledgment when submitting your source files to exercises. In addition, you are not allowed to collaborate on the exams.
For doing PAs, you can discuss ideas, concepts, and code, taken from the lab exercises, the book, or the web, with your peers, mentor, and instructor. Note that solutions to PAs must be your own work. Any help you receive from the above mentioned sources for solving the programming assignments must be fully acknowledged. To do this, please include comments in your source codes that specifically state original sources (e.g., people or web addresses) of your copied or modified codes.
If your solutions to the programming assignments indicate academic dishonesty, the Committee on Academic Standing will deal with them. Please read notes on the academic honesty, determined by the Grinnell College and the Department of Computer Science, in the following links:
https://catalog.grinnell.edu/content.php?catoid=12&navoid=2537#Honesty_in_Academic_Work
https://www.cs.grinnell.edu/academic-honesty-policy.html

Accommodation
Grinnell College provides reasonable accommodations if you have documented disabilities. To discuss your needs and provide required documentation, please contact the Coordinator for Student Disability Resources, John Hirschman (hirschma@grinnell.edu), located on the 3rd floor of Goodnow Hall (x3089). After contacting the Student Disability Resources, please notify me, as soon as you can, to ensure your full participation in the course and coordinate your accommodations.

Religious Observance Policy
Grinnell College acknowledges and embraces the religious diversity of its faculty, students and staff. Faculty and students share responsibility to support members of our community who observe religious holidays. Students will provide faculty members with reasonable notice of the dates of religious holidays on which they will be absent, and this notice would be expected to occur no later than the third weeks of the semester. Faculty members will make reasonable efforts to accommodate students who need to be absent from examinations or class due to religious observance. Students are responsible for completing any part of the course work, including examinations, they have missed due to religious observance, and faculty members are responsible for giving them the opportunity to do so.

Academic Alerts
Because I value your success as a student in my course, I may reach out to our Academic Advising staff if I notice that you need additional support to succeed in this course. I will submit an academic alert using the college’s SAL system. Before submitting the alert, I will do my best to notify you, but may need to act quickly depending on circumstances. With the academic alert, the College invites one or more trained professionals on campus to work in partnership with you and me to address issues submitted in the alert. The goal is to ensure your success in the course and at Grinnell. You are always welcome to reach out to me about issues that impact your learning, and I will be happy to help you connect directly with campus resources.