Course Name: Data Structures

Course Number: CS 261

Course Credits: This course combines approximately 120 hours of instruction, on-line activities, exams, and assignments for 4 credits.

Instructor Info: Randy Scovil / randy.scovil@oregonstate.edu

Teaching Assistants' names and contact info:

- TAs and Office Hours Information on Course Syllabus Page (in Canvas)

Textbook: None

Course Description

Abstract Data Types, Dynamic Arrays, Linked Lists, Trees and Graphs, Binary Search Trees, Hash Tables, Storage Management, Complexity Analysis of Data Structures.

Prerequisites:

(CS 162 or CS 165) and (CS 225 or MTH 231) [C or better in each course]

Communication

1. Please post all course-related questions in the Ed Discussions forum so that the whole class may benefit from your conversation. Not all posts require a reply from the instructor/TA and often it is better for students to hash out an answer to a question.

2. Office hours will be held online through the class Microsoft Teams Team. To get started on Teams please go to https://is.oregonstate.edu/teams. Be sure to add the class-specific Teams – directions are in the first course Module.

3. Please email (do NOT use Canvas Messaging) your instructor only for matters of a personal or private (grading) nature. The instructor or a TA will reply to most course-related questions within 48 hours, though this may vary on weekends.

4. Any email sent to the instructor about this course **must** originate with an OSU supplied email account *and contain the tag* **[CS261]** at the beginning of the subject. Failure to comply with this will result in a delayed (or possibly nonexistent) response to your email.

Canvas & Ed Discussions

1. This course will be delivered via Canvas, and you will interact with your classmates and with your instructor through Ed Discussions. Within the course Canvas site, you will access the learning materials, such as the syllabus, modules, assignments, and exams. Class discussions will be on Ed Discussions.

2. To preview how an online course works, visit the [Ecampus Course Demo](). For technical assistance, please visit [Ecampus Technical Help]().

3. Canvas is optimized for the most recent versions of most popular browsers. If your browser of choice is an out-of-date version, you should update it for use with Canvas, especially for exams. If you are having browser troubles, seek out the Technical Assistance described below.

4. If your primary device has trouble dealing with Canvas (e.g., as some tablets do), make sure that you have an alternative available for things like exams. If you are having device troubles, seek out the Technical Assistance described below. It is your responsibility to make sure that you have a trouble-free device before taking the exam.

5. General announcements for the class will be sent as emails through Canvas and posted as Canvas announcements. It is your responsibility to keep up with messages in Canvas Announcements and Inbox Messages.

Technical Assistance

If you experience errors or problems while in your online course, contact 24-7 Canvas Support through the Help link within Canvas. If you experience computer difficulties, need help downloading a browser or plug-in, or need help logging into a course, contact the IS Service Desk for assistance. You can call (541) 737-8787 or visit the [IS Service Desk]() online.

Course Learning Objectives

At the completion of the course, students will be able to...

1. **Describe** the properties, interfaces, and behaviors of basic abstract data types, such as collection, bag, indexed collection, sorted collection, stack, and queue.

2. **State** the asymptotic time complexity of the fundamental operations associated with a variety of data structures, such as vector, linked list, tree, and heap.

3. **Recall** the space utilization of common data structures in terms of the long-term storage needed to maintain the structure, as well as the short-term memory requirements of fundamental operations, such as sorting.

4. **Design** and **implement** general-purpose, reusable data structures that implement one or more abstractions.

5. **Compare** and **contrast** the operation of common data structures (such as linear structures, priority queues, tree structures, hash tables, maps, and graphs) in terms of time complexity, space utilization, and the abstract data types they implement.

6. **Read** an algorithm or program code segment that contains iterative constructs and **analyze** the asymptotic time complexity of the algorithm or code segment.

Evaluation of Student Performance

   70% – Assignments (programming and written)

   29% – Exams (midterm – 12% and final exam – 17%)

   1% – Syllabus Quiz

Letter Grade Scale:

Grade
100 >= A >= 92.5
92.5 > A- >= 89.5
89.5 > B+ >= 86.5
86.5 > B >= 82.5
82.5 > B- >= 79.5
79.5 > C+ >= 76.5
76.5 > C >= 72.5
72.5 > C- >= 69.5
69.5 > D+ >= 66.5
66.5 > D >= 62.5
62.5 > D- >= 59.5
59.5 > F

*** REMINDER: A passing grade for core classes in CS is a C or above. A C- (below 72.5) is not a passing grade for CS majors.

Assignments (70%)

- Assignments will primarily consist of writing computer programs and some assignments will require written answers to questions.

- Assignments are to be turned in **before 23:59 (PST)** on the date they are due.

- Assignments will be turned in via Canvas or Gradescope. Typically, written assignments are submitted to Canvas and programming assignments are uploaded to Gradescope. Assignment 0 will provide you with an opportunity to become familiar with Gradescope and submitting assignments there.

- In order to receive full marks your program must pass all the Gradescope tests while adhering to the provided specifications. If a grader suspects that a submitted program has been written in such a way that it passes a listed grading test on Gradescope, but does not adhere to the specification, then an additional test is likely to be run. However, any additional test done shall be easily justified as checking that the program adheres to the specification. The number of points to be awarded or deducted by these additional tests is at the discretion of the grader.

- The most recent Gradescope submission will be graded unless a different one is activated by the student. It is the student's responsibility to do so before the deadline.

- Any crashes, hangs, errors, infinite loops, etc. not covered in the grading instructions and/or grading scripts will cause your program to lose points on Gradescope. If your program does not work on Gradescope, you are responsible for identifying which function/s that is/are causing the failure and then filing a Regrade Request on Gradescope for a regrading for the rest of your submission. The points lost depend on the severity, how much it affects the rest of the program, and how it is recovered from, if at all, all based on the discretion of the grader.

- All source files (`.py` files) must include a comment block at the top that contains your name and OSU email address, the class number/section, the assignment name, due date, and program description in your own words. Neglecting this information will result in a point loss on any assignment.

- Assignment code may be visually inspected by the grader to verify compliance with assignment specifications.

- **We reserve the right to ask you to explain a complicated piece of code**. If you cannot explain your own code to us, you may be investigated for violation of student conduct.

- If you are struggling on a programming assignment, the first thing you should do is make sure you have worked through the assigned and relevant content and reviewed the notes you've taken.
  - For best results when asking for help, provide as much information as possible (if you don't, we'll ask you for it.) Part of your job is to attack what the problem might be (and what it isn't.)
    - Add context to your message – what's wrong, what you suspect it might be, as well as anything you've already tried and ruled out.
  - Run the program in the PyCharm debugger yourself. If you are unfamiliar with debugging in PyCharm, refer to the Coding Guides and Tips module provided in Canvas.
  - Make sure you read the entire assignment specification. The specifications are there for a reason and often provide insights into an optimal solution.

- Any failure to follow specifications (e.g., submitting the wrong type of file) may result in a deduction at the discretion of the grader.

- If you wish to petition a grade, you must do so within one week of receipt of the grade, by filing a Regrade Request on Gradescope.

Exams (29% Total)

- There are 2 exams (other than the syllabus quiz) in this course.

- Please check (and save!) the actual dates for the Midterm and Final provided in the **Course Schedule** document and on Canvas. You will have a 5-day time window to take each exam. No extension will be allowed outside those windows.

- Exams only test knowledge of the course material, not Stack Overflow or anywhere else on the Internet.

- The exams are timed, closed-book, closed-notes, and not proctored. Please be aware that multiple attempts of the exam are not allowed. Be sure to take the exam right away even if you open the exam accidentally.

- Be sure to check Canvas ahead of time for the assigned time limit for each test. Both tests will likely have a time limit of approximately 120 minutes, but this may vary.

- If you wish to petition a grade, you must do so within one week of receipt of the grade by posting a note on your exam page in Canvas.

- **Finally, no late submission for the exams will be graded.** Plan ahead to take and submit the exams before the deadline. Please don't make any requests for exam deadline extensions.

Late Work Policy

If you are adversely impacted by circumstances outside of your control and it impedes your ability to complete your work done on time, please email the instructor to let them know about any extenuating circumstances as early as possible.

- Assignments may be turned in a maximum of two days late. Late assignments are penalized 15% (of the earned points) for each day or portion of a day they are late.
- You will be provided with 4 "free" late days that you can apply towards any assignment, but you still must turn the assignment in no more than two days late.
- Each "free" day applied will remove one day's penalty from that assignment, up to a maximum of two days for that assignment.
- You must communicate your intent to use a free day (or two) *before the assignment is due (i.e., the original assignment due date)*, and via the following process:
    o Email the instructor (via the email guidelines above) and tag the subject **[CS261 – Using Free Days for Assignment X]**, where X is the appropriate assignment number.
    o In the email specify the number of days you plan to use. If you declare that you will use two "free" days but only end up needing one, you will burn only one of your days if you send a follow-up message to update your request and send the follow-up request before the two-day late period has expired.
- Any assignment that is submitted after the two-day late window will not be graded and will receive a score of 0.

Incompletes

In this online program, there will rarely be cases where an incomplete is appropriate. The instructor will only consider giving an incomplete grade for emergency cases such as a death in the family, major disease, or childbirth, while also having completed at least 60% of all coursework. If you have a situation that may prevent you from completing the coursework, let the instructor know as soon as you can.

Academic Honesty

- Students are expected to do their own work. The only sources you're allowed to use code from are the explorations, and no citation of this code is necessary. Direct use of any other resources is prohibited.

- Programming assignments present unique challenges for graders. It is often difficult for a grader to distinguish between legitimate help and plagiarism. We use plagiarism-detection software to check your code against the code from other students. It is quite sophisticated and can see through variable name changes and formatting differences.

- Honesty is essential for learning to take place. It will form the foundation of your professional integrity in your career.

- Specific Examples:

    - You **may** ask conceptual questions related to optimizing your code on Teams or Ed.

    - You are **encouraged to** discuss course content with other students, TAs, the instructor, or anyone else who will listen, including general discussion of homework assignments and how to fix specific issues.

    - You **may** share pseudocode and ideas about how to solve or approach problems. If you are getting odd error messages, you can share the snippet of code that is producing the message; do not share the entire file.

    - You **may not** post any complete functions/procedures/logic blocks to Teams or Ed.

    - You **must** write new, original programs, even if you are re-taking the course.

        - If you are retaking the course, you **must** generate new programming solutions (you **may not** re-use your own work from previous terms)

    - You **may not** copy anyone's (or allow someone to copy your) solutions. It is possible to discuss problems without plagiarizing. One of the best methods of debugging is to explain your solution to someone else.

    - You **may not** post any exam questions or solutions in any form.

    - You **must** make any git repo you post this code on private, with the exception of the Portfolio assignment.

- If you are found in violation of any of the above policies, whether you are the giver or receiver of help, you will be subject to the University Academic Misconduct process. The first offense usually results in a warning and an assignment penalty (0-grade); the second offense can result in a disciplinary hearing, and possibly removal from the Program.

- The bottom line is: Each student is expected to understand all aspects of the programs they submit for credit.

- The following are examples of plagiarism, drawn from actual submitted and penalized cases.

- Student Googled the problem and ended up finding a code written online. Student used the code there to improve their own work.

- Student worked with a classmate, each tackling one half of the program. Both students modified the identifiers, labels, comments, etc...

- Student found a solution to the problem online and changed a few variable names and methods and handed it as their own.

Academic Integrity

- The Code of Student Conduct prohibits Academic Misconduct and defines it as:
  - Any action that misrepresents a student or group's work, knowledge, or achievement, provides a potential or actual inequitable advantage, or compromises the integrity of the educational process.
- To support understanding of what can be included in this definition, the Code further classifies and describes examples of Academic Misconduct, including cheating, plagiarism, assisting and others. See the Code of Student Conduct for details.
- You are expected to do your own work and demonstrate academic integrity in every aspect of this course. Familiarize yourself with the standards set forth in the **OSU Code of Student Conduct Section 4.2.** You must only access sources and resources authorized by the instructor. You may not show your work to any other current or future students without the instructor's authorization. Violations of these expectations or the Code of Student Conduct will be reported to the Office of Student Conduct and Community Standards. If there is any question about whether an act constitutes academic misconduct, it is your responsibility to seek clarification and approval from the instructor prior to acting.

Statement Regarding Religious Accommodation

Oregon State University is required to provide reasonable accommodations for employee and student sincerely held religious beliefs. It is incumbent on the student making the request to make the faculty member aware of the request as soon as possible prior to the need for the accommodation. See the Religious Accommodation Process for Students.

Guidelines for a Productive and Effective Online Classroom
(Adapted from Dr. Susan Shaw, Oregon State University)

Students are expected to conduct themselves in the course (e.g., on discussion boards, email) in compliance with the university's regulations regarding civility. Civility is an essential ingredient for academic discourse. All communications for this course should be conducted constructively, civilly, and respectfully. Differences in beliefs, opinions, and approaches are to be expected.

In all you say and do for this course, be professional. Please bring any communications you believe to be in violation of this class policy to the attention of your instructor.

Active interaction with peers and your instructor is essential to success in this online course, paying particular attention to the following:

- Unless indicated otherwise, please complete the readings and view other instructional materials for each week before participating in the discussion board.

- Read your posts carefully before submitting them.

- Be respectful of others and their opinions, valuing diversity in backgrounds, abilities, and experiences.

- Challenging the ideas held by others is an integral aspect of critical thinking and the academic process. Please word your responses carefully and recognize that others are expected to challenge your ideas. A positive atmosphere of healthy debate is encouraged.

Expectations for Student Conduct

Student conduct is governed by the university's policies, as explained in the Student Conduct Code (https://beav.es/codeofconduct). Students are expected to conduct themselves in the course (e.g., on discussion boards, email postings) in compliance with the university's regulations regarding civility.

Personal Issues Conflicting with Course Progression

Your advisors are excellent resources when significant personal issues conflict with your ability to progress in the course. Remember to keep them (and us) updated in these situations.

Academic Calendar

All students are subject to the registration and refund deadlines as stated in the Academic Calendar: https://registrar.oregonstate.edu/osu-academic-calendar

Statement Regarding Students with Disabilities

Accommodations for students with disabilities are determined and approved by Disability Access Services (DAS). If you, as a student, believe you are eligible for accommodations but have not obtained approval, please contact DAS immediately at 541-737-4098 or at http://ds.oregonstate.edu. DAS notifies students and faculty members of approved academic accommodations and coordinates implementation of those accommodations. While not required, students and faculty members are encouraged to discuss details of the implementation of individual accommodations.

Course Content

Module	Topics	Module	Topics
1	• Introduction to Data Structures	6	• AVL Trees and Balancing • AVL Tree Rotations
2	• Big O Notation • Binary Search	7	• Priority Queues • Heaps
3	• ADT Overview • Python's List • Iterators	8	• Maps • Hash Tables • Hash Table Collisions
4	• Linked Lists • Stacks, and Queues	9	• Graphs
5	• Trees • Binary Trees • BST Operations	10	• Advanced Sorting

Accessibility of Course Materials

All materials used in this course are accessible. If you require accommodations, please contact Disability Access Services (DAS).

Additionally, Canvas, the learning management system through which this course is offered, provides a vendor statement certifying how the platform is accessible to students with disabilities.

Tutoring and Writing Assistance

The OSU College of Engineering has in-College tutors available for Computer Science students. To sign up for this service, please visit the OSU CoE Academic Support site.

The Oregon State Online Writing Suite is also available for students enrolled in Ecampus courses.

Ecampus Reach Out for Success

University students encounter setbacks from time to time. If you encounter difficulties and need assistance, it's important to reach out. Consider discussing the situation with an instructor or academic advisor. Learn about resources that assist with wellness and academic success.

Ecampus students are always encouraged to discuss issues that impact your academic success with the [Ecampus Success Team](#). Email [ecampus.success@oregonstate.edu](#) to identify strategies and resources that can support you in your educational goals.

- **For mental health:**

  Learn about [counseling and psychological resources for Ecampus students](#). If you are in immediate crisis, please contact the Crisis Text Line by texting OREGON to 741-741 or call the National Suicide Prevention Lifeline at 1-800-273-TALK (8255).

- **For financial hardship:**

  Any student whose academic performance is impacted due to financial stress or the inability to afford groceries, housing, and other necessities for any reason is urged to contact the Director of Care for support (541-737-8748).

Establishing a Positive Community

It is important you feel safe and welcome in this course. If somebody is making discriminatory comments against you, sexually harassing you, or excluding you in other ways, contact the instructor, your academic advisor, and/or report what happened at https://studentlife.oregonstate.edu/studentconduct/reporting so we can connect you with resources.

Student Evaluation of Courses

The online Student Evaluation of Teaching system opens to students during the week before finals and closes the Monday following the end of finals. Students receive notification, instructions, and the link through their ONID. They may also log into the system via Online Services. Course evaluation results are extremely important and used to help improve courses and the online learning experience for future students. Responses are anonymous (unless a student chooses to "sign" their comments, agreeing to relinquish anonymity) and unavailable to instructors until after grades have been posted. The results of scaled questions and signed comments go to both the instructor and their unit head/supervisor. Anonymous (unsigned) comments go to the instructor only.

Student Bill of Rights

OSU has twelve established student rights. They include due process in all university disciplinary processes, an equal opportunity to learn, and grading in accordance with the course syllabus: https://asosu.oregonstate.edu/advocacy/rights

Potential Updates to this Syllabus During the Course

Please note: The instructor reserves the right to make changes to this syllabus as needed. All changes will be communicated via email and/or Canvas announcements.