Fall 2018 CS 165 syllabus

Course: CS 165 – Accelerated Introduction to Computer Science

Credits: 8

Instructor: Anahita Sanandaji

Email: sanandaa@oregonstate.edu

Two fundamental rules

1. You are responsible for knowing the contents of the syllabus and all of the information about the course provided on Canvas.
2. You are responsible for knowing the contents of any instructor emails sent to you, instructor messages sent to you via Canvas or Piazza, and instructor announcements made on Canvas, which means that you should make sure you receive such communications, that you check for new ones at least once a day, and that you read them. (You do not have to read all posts anyone makes in Piazza, or even all the posts I make in Piazza - just the ones that are in response to a post you made.)

OSU catalog course description, including pre-requisites/co-requisites

Overview of fundamental concepts of computer science. Introduction to problem solving, software engineering and object-oriented algorithm development and programming. Lec/lab. PREREQS: MTH 111 [C] or Placement Test MPAL(060) and CS Double Degree students must have a BA/BS degree.

Course content

- identifiers and primitive data types
- assignment, arithmetic, logical and relational operators
- expressions and statements
- debugging
- flow of control: selection, repetition
- functions, parameter passing, call by value and call by reference
- object-oriented programming, polymorphism, operator overloading
- one- and two-dimensional arrays, strings and other structured data types
- pointers

- recursion
- searching, sorting, big-O
- operator overloading, inheritance, polymorphism
- exceptions, templates
- basic data structures

Course Learning Outcomes

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

1. **Design** and **implement** programs that require
   - various control statements involving selection and repetition
   - expressions with variables, constants, function calls, pointers, and arithmetic/relational operators with mixed data
   - arrays, strings, and other data structures
   - library functions and programmer-defined functions with parameter-passing by value and by reference
2. **Define** and **use** classes and objects
3. **Debug** programming syntax and run-time errors.
4. **Produce** recursive algorithms
5. **Describe** and **apply** basic software engineering design principles and software quality factors.
6. **Design** and **implement** programs that require:
   - multiple classes and structures
   - hierarchies of classes that use inheritance and polymorphism
   - understanding of abstraction, modularity, separation of concerns, exception handling
7. **Construct** and **use** basic linear structures (arrays, stacks, queues, and various linked lists) in programs, and be able to describe instances appropriate for their use.
8. **Classify** moderately complicated algorithms in these complexity classes: O(1), O(log n), O(n), O(n log n), and O(n^2).
9. **Develop** test-data sets and testing plans for programming projects
10. **Produce** recursive algorithms, and **choose** appropriately between iterative and recursive algorithms.

Textbook (required)

C++ Early Objects, 9th ed. by Gaddis et al.

Course tools

- Canvas is the course management software used for this course.
- TEACH is the website where you will enable your ENGR account so you can log in to the school server ("flip").
- Ecampus Exams and Proctoring Form is where you will tell us who your proctor will be for the exams.
- PuTTY/Terminal are terminal emulators – they provide a window where you will interact with the OSU server (flip), using a command-line interface. Terminal is built into Macs. For Windows you can download and install PuTTY.
- Linux is the operating system used on flip.
- scp/FileZilla are ways you can transfer files between your computer and flip. You can use the scp command in the command line of your terminal emulator, but FileZilla provides a convenient graphical interface.vim/nano/Xcode/Visual Studio are examples of code editors that provide an environment in which to create and edit computer programs.
- Mimir is the site where you will submit your code for the assignments.
- Piazza is an asynchronous Q&A discussion forum where you can get official answers from the instructor.
- Slack is a synchronous discussion forum for more casual conversation with fellow students.
- More information about these tools is available in the "Start Here" module on Canvas.

Proctoring

There will be two proctored exams in this course. For each exam you will have a week during which to schedule it. You will be responsible for finding a proctor, submitting your proctor info to Ecampus, and scheduling a time for your exam (within the allotted week). If your computer runs the Linux operating system (instead of Mac or Windows), then you cannot use Proctor U.There is more information about proctoring in the Proctored Exam Information module.

Coursework and Grading Policies

- For accurate results, you must test your code on the school server ("flip"). It is possible for code to compile and run correctly in another environment, but not on the school server. Your code will be tested on Mimir, which uses the same compiler that is used on flip.
- If your program doesn't compile on flip, using the specified file names, it will get a zero, even if it works perfectly in some other coding environment.

- Your code will be tested to make sure it works correctly. It is your job to make sure that your code will behave correctly for any possible tests. I provide one simple test on Mimir for which you get immediate feedback, so you can verify that your code is compiling and there aren't any formatting issues. The remaining tests are "hidden" until the due date has passed.
- Late submissions are not graded. It is better to submit an incomplete (but compiling) program on time for possible partial credit than to not have your program graded at all.
- Be aware of what time zone Canvas is using to display deadlines. If it's not your local time zone, you can set it to be. The Canvas phone app always displays times for the time zone the phone is in. This is important to keep in mind if you will be traveling.
- Always ask first before using any commands or constructs that we haven't covered yet, otherwise you may lose points. The main reason for this is that I don't want you to circumvent things the assignment was designed to make you do.
- If you disagree with a score on any coursework, contact your TA by email within one week of receiving your grade. If you are dissatisfied with your TA's response (or if they neglect to respond at all), then you should contact me about it.
- There is no extra credit.
- I don't plan to “curve” the grades, but that's something I'll re-assess at the end of the term.
- Makeup exams take considerable effort to schedule, so they will not be given under normal circumstances. If you learn about an event that may cause you to alter your exam scheduling, then contact me and your proctor (or the testing coordinator) as soon as you can so that accommodations may be considered.
- Incompletes will be given very rarely. If you have been doing well in the course so far, but an emergency comes up that prevents you from continuing according to schedule, let me know as soon as you can.

Weights for Grading

Assignments	60%
Exams	40%

Letter Grade Percentages

93-100%	A	80-83%	B-	67-70%	D+
90-93%	A-	77-80%	C+	63-67%	D

87-90%	B+	73-77%	C	60-63%	D-
83-87%	B	70-73%	C-	0-60%	F

Being Mentally Prepared

Learning how to program a computer can be quite challenging for most people. That is true even in CS 161, and CS 165 goes twice as fast. You should expect to spend 32-40 hours/week. However, that is only an estimate. Some people will find the material more challenging than others - those people may require more time.

Other people in the course may have more background in the subject than you do. Don't feel intimidated or put off if other students talk on the discussion board about topics that we haven't covered yet (or may not cover in this class at all). What's important is that you understand what we have covered.

Taking online courses tends to feel more isolated than taking on-campus courses. Even though there are still all the normal course elements - lectures, readings, homework, a teacher whom you can ask questions, etc. - sometimes students in an online course will feel like they are "teaching themselves" because interactivity in such a course is less forced and less immediate. I can't see or hear you, so I can't judge whether you're having trouble by your tone or expression. I am always happy to help, but you have to take the initiative and let me know when you run into problems.

Getting Help

When you have a question about something, the order of steps you should take to pursue answers is roughly the following:

1. Review the relevant materials (assignment description, readings, lectures).
2. Search the class discussion board on Piazza (it can take some practice to learn how to refine your searches well).
3. Post to the class discussion board on Piazza (by doing #1 and #2 first, you might save yourself asking a question, and you might also then be better able to help others). When you post a question on Piazza and it's answered there, that benefits other people who have the same question (if you're shy, you can post anonymously).
4. If you've tried #1-3 and feel like your question hasn't been fully addressed, please email your assigned TA or myself.

You are allowed to post to the discussion board any non-working code you need help with, but you should post only as much as is necessary for your question. Do not post working code until after the assignment due date. When answering a posted question, please use pseudocode or give hints so the student will have the satisfaction (and learning reinforcement) of figuring out the solution for themselves. When you are willing to help others on the discussion board and take pains to not make anyone feel like they asked a dumb question, you are reinforcing both your technical knowledge and your people skills, both of which are highly valued not just here, but out in the real world.

Academic Integrity

For this class, it is encouraged for students to discuss course content with each other, even including general discussion of homework assignments and how to fix specific issues. However, each person must develop her or his own individual solutions (except of course in group assignments, where each group must develop its own solutions). In particular, a student may not copy (by any means) another's work (or portions of it) and represent it as her/his own. Plagiarism can result in drastic consequences for both the person who copied and the person who allowed them to copy. These may include a zero for the assignment or failure from the course. **If you make your code publicly accessible (such as in a public github repository), that qualifies as plagiarism.**

Students with Disabilities

Accommodations are collaborative efforts between students, faculty, and Disability Access Services (DAS). Those with accommodations approved through DAS are responsible for contacting the faculty member in charge of the course prior to or during the first week of the term to discuss accommodations. Students who believe they are eligible for accommodations but who have not yet obtained approval through DAS should contact them as soon as possible (541-737-4098).

Student Conduct Code

This program strives to prepare students for careers in computer science, which includes preparing students to communicate professionally. Therefore, students in this class are expected to communicate in a professional manner in discussion forums, email messages and all communications for this course. Critiques, disagreements, problems, or other topics of a sensitive nature can be addressed, but should be addressed civilly and professionally. If a student's communications become unprofessional, disruptive, abusive, inflammatory, or if they otherwise obstruct the learning process of the class, the instructor may restrict the student from participating in the electronic forums associated with the class and notify Ecampus and the OSU Office of Student Conduct and Community Standards. Productive learning communities and workplaces depend on civil, professional discourse. It is our hope that this policy strengthens your learning community and prepares you for the professional workplace.

Online Privacy

Posts to the discussion board are public messages, and all writings in this area will be viewable by the entire class or assigned group members. If you prefer that only the instructor sees your communication, send it via private message or email.

Posting of personal contact information is strictly at your own risk.