Computer Science & Engineering 142: Introduction to Programming I
Course Syllabus, Winter 2020

Instructor	Brett Wortzman	
Email	brettwo@cs.uw.edu	
Office	CSE 542	
Office Hours	Tuesdays 2:00-3:30pm	
	Wednesdays 4:00-5:30pm	
Course Website	https://cs.uw.edu/142	All course materials and resources will be posted here. Check this site frequently for updates.
Lectures	A Lecture: MWF 11:30am – 12:20pm, KNE 130	
	B Lecture: MWF 2:30pm – 3:20pm, KNE 120	
Course Administrator	Pim Lustig	Pim is your best resource for registration questions, such as switching sections or changing to or from S/NS grading.
Email	cse142@uw.edu	
Office	CSE2 173	

Course Overview/Objectives

This course provides an introduction to programming using the Java programming language. We will explore common computational problem-solving techniques useful to computer scientists, but also to anyone who has large data sets, repetitive processes or other needs for computation. No prior programming experience is assumed, although students should know the basics of using a computer (e.g., using a web browser and a text editor) and should be comfortable with math through Algebra 1. Students with significant prior programming experience should consider skipping CSE 142 and taking CSE 143 (we allow this without any special permission) or CSE 143X.

At the end of this course, students should be able to:

- produce functional, well-written Java programs of small to medium length and complexity
- utilize a variety of programming constructs (including, but not limited to, methods, loops, conditionals, arrays, and classes) to solve problems
- explain the importance of code that is not just functional, but well-written, readable, and maintainable
- identify and fix bugs and errors that occur during the development process
- identify and utilize resources to help overcome difficulties or resolve errors in developing programs

Course Components

Lectures

Lectures will be the place where you are first introduced to new concepts. Although we do not take attendance in lecture, all students are expected to attend lecture each day and will be held responsible for all material (including administrative announcements) presented. All lectures will be recorded and posted on the course website within 24 hours of the last lecture on that day. If you must miss lecture, you should view the recording and/or speak to your TA or instructor about what you have missed.

It is not expected that will have mastered the material by the end of lecture because mastery requires practice. To encourage students to participate actively in their learning during lectures, we will be using PollEverywhere polls in class. These polls are an opportunity to solve problems that help you check your understanding and are **not graded**.

Students are expected to keep talking and electronics usage to a minimum during lecture. You are welcome to use a laptop or tablet to take notes or code along, but please remain on task. If you expect to use your device in ways not related to the course, please sit only in the last four rows of the classroom. This is to ensure a productive learning environment for all the students in the classroom. TAs will periodically enforce this policy during lecture, and students acting in a disruptive or disrespectful manner may be asked to leave the classroom.

Discussion Sections

All students are assigned to a discussion section, held at various times on Thursdays. These sections are led by one of our TAs, and provide opportunity to reinforce and practice concepts covered in lecture in a smaller group setting. In section, we will answer questions, complete and discuss sample problems, and provide tips for that week's topics and assignment (see below). The TA who leads your section will also grade your programming assignments.

Attendance and participation in section is mandatory and is part of your course grade (see below). You must attend the section to which you are assigned to receive credit. If you need to attend a different section for a specific week, you must contact your TA ahead of time and receive permission. You should *not* regularly attend a section for which you are not registered. (Contact the course administrator if you need to change your section.)

Grading and Assessment

Grades for CSE 142 are calculated based on three categories, weighted as follows:

45%	programming assignments, written reflections, and section participation	
20%	midterm exam	Friday, February 14th (tentative)
35%	final exam	Wednesday, March 18th (tentative)

Final course grades are converted to the 4.0 scale roughly as follows:

90%: at least 3.5	80%: at least 2.5	70%: at least 1.5	60%: at least 0.7

Programming Assignments

There will be eight (8) weekly programming assignments, which must be completed individually and submitted electronically via the course web site. Assignments will be graded on both "external correctness" (behavior) and "internal correctness" (adherence to design, usage, and style guidelines) and are generally graded on a 20-point scale.

Reflections

There will be eleven (11) total reflection assignments over the course of the quarter: one post-reflection for each homework assignment (eight total), a pre-reflection for each exam (two total), and a post-reflection for the midterm exam. Reflections are short written assignments that will help you engage in metacognition and assess the strengths and weaknesses in both your understanding and your process. These assignments will be graded entirely on effort and completion, and will not be accepted late for any reason. You will earn 2 points per completed reflection, up to a maximum of 20 points for the quarter.

Section Participation/Homework

Two points are awarded for being present and participating in section each week. An additional point is awarded for completing short homework problems that are posted on the course website and due at the **start** of each week's section. Section homework is expected to take no more than 60 minutes each week and is graded entirely on effort and completion. Section homework must be submitted physically in section each week, and will not be accepted late for any reason. You will earn 2 or 3 points per section attended, up to a maximum of 20 points for the quarter.

Exams

There will be two exams, a midterm and final, both of which will be closed-book and closed-note. You may not bring any written material, but a standard "cheat sheet" will be provided as part of the exam. No electronic devices, including calculators or smart devices, may be used.

Late Work

Each student receives **five (5) "late days"** for use on programming assignments. A late day allows you to submit an assignment up to 24 hours late without penalty. (For example, you could use 2 late days and submit an assignment due at 11:59pm on Tuesday by 11:59pm on Thursday with no penalty. Once a student has used up all their late days, each successive day that an assignment is late will result in a loss of 1 point on that assignment. Regardless of how many late days you have, **you may not submit an assignment more than three (3) days after it is due** or after the last day of class. It is *highly recommended* that you save late days for unexpected situations and not use them too early in the quarter.

Late days may only be used for programming assignments; they may not be used for reflections, section homework, or exams.

Make-up Work/Extensions

In general, all students are expected to complete all work on time and as scheduled. For programming assignments, students are expected to utilize their late days to deal with situations in which they must turn in work late. Students with a known conflict (e.g. another class, family responsibilities, work that cannot be rescheduled) with a scheduled exam must request an alternate by the deadline and via the process posted on the course website. Requests for extensions or make-ups for known conflicts will not be accepted after the posted deadline.

In the event of an unexpected or extenuating circumstance (including illness or injury), students must contact the instructor (or arrange for someone else to do so) **as soon as possible**, **before** the scheduled assignment "lock date" or exam, to explain the situation. These situations will be handled on a case-by-case basis. Note that the sooner you reach out to explain the situation, the more likely will be able to reach a resolution.

Under no circumstances will programming assignments be accepted after scores for that assignment have been returned. No extensions or alternate exams will be granted for personal reasons (e.g. travel, social plans).

Course Policies

Getting Help

Learning programming for the first time can be very challenging, and it is expected that you will struggle over the course of the quarter. Please don't be afraid to ask for help if you don't understand something or need assistance with assignments or material. There are several resources available to support you in the course

The **course message board (Piazza)** is available to ask questions of the course staff, both publicly and privately. Whenever possible, please prefer posting a public question, as many of students likely have the same question and will benefit from seeing the response. Private questions should be reserved for personal matters or cases where you must post your code (see below). Please make an effort to search for a question similar to yours before posting; this will help reduce load on the course staff and ensure we can answer all questions efficiently and effectively. Programming assignment code should *never* be included in a public message board post. However, you are welcome to post your programming assignment code privately, or to post code not related to the programming assignments (e.g. from a practice problem or lecture/section example).

The **Introductory Programming Lab (IPL)** is located in **room 334 of Mary Gates Hall**. TAs will be available at the lab to help students with problems. It is open almost every day of the week; the actual schedule is available on the course website. In addition to seeking help on programming assignments, you can ask for a review or explanation of lecture material, support on a practice problem, or general advice on how to succeed in the course. TAs in the IPL will *not* tell you exactly how to complete an assignment or fix a bug you may be having, but they will be happy to help you determine steps *you* can take to solve the problem.

Brett holds **office hours** multiple times a week—see the course website for specific times. Office hours are a great time to ask for clarification on a topic or discuss course logistics. No appointments are necessary for general questions, but please note that office hours are open door and privacy cannot be guaranteed. If you have a personal question that you would prefer to discuss without other students around, please request a separate appointment.

Inclusion

All students are welcome in CSE142 and are entitled to be treated respectfully by both classmates and the course staff. We strive to create a challenging but inclusive environment that is conducive to learning for all students. If at any time you feel that you are not experiencing an inclusive environment, please contact the course staff or the CSE academic advisors. You should feel free to email any member of the course staff or the advisors at any time, and anonymous feedback can be sent to the course staff via the form linked on the course website.

Accommodations

Please refer to university policies regarding accommodations for purposes of disability or religion. Accommodations must be formally requested through the proper channels according to the relevant policies.

Academic Integrity and Collaboration

Students are encouraged to utilize all available resources, including each other, in developing mastery of the course content. You are more than welcome to work with other students on in-class polls, section problems, practice problems, labs, or any other supplementary material. You are also encouraged to work with classmates to prepare for exams and to understand the goals and requirements of programming assignments at a high level. However, our ultimate goal is to ensure that *each individual student* learns the course material so that you will be prepared for future courses, projects, jobs, research, etc.

To that end, **all programming assignments MUST be completed individually, and all code you submit must be ENTRIELY your own work**. In particular, violations of this policy include, but are not limited to:

- Working as a partner with another student on an assignment and submitting work completed collaboratively.
- Showing another student your solution to an assignment, or looking at a solution from another student (past or present), for any reason.
- Having another person "walk you through" an assignment, describe in detail how to solve it, or sit with you as you code; or providing such help to another student.
  - This includes current or former students, tutors, friends, TAs, paid consultants, people on the Internet, or anyone else.
- Copying *any amount* of code (even just a few lines or a single method) from anywhere other than resources explicitly allowed by the instructor.
- Posting your homework solution code, in part or in whole, online to ask others for help.
  - This includes public message boards, code repositories, forums, file sharing sites and services, or any other online system.

Note that rewriting someone else's code yourself, making changes to existing code, or being able to thoroughly explain code you did **do not** constitute making the work your own. Any support or assistance you receive from individuals outside of the course staff must be at a high level and not include any code or detailed descriptions of code. Referencing or utilizing code not provided by the course in *any way* is likely a violation of this policy.

Under our policy, a student who gives inappropriate assistance is equally guilty as one who receives it. That is, you will be considered to have violated this policy if you provide your code to another student, even if you had initially written the code yourself. Instead of providing such help, please refer others to course resources such as lecture examples, the textbook, the IPL, or a TA or instructor. **You must not share your solution with others under any circumstances.** In addition, be sure to protect your code by logging out of shared computers, not leaving printouts of your code in public places, not emailing your code to other students, and not storing your code in publicly accessible places, such as public places on the web or shared computers or cloud drives.

If you are retaking the course, you may resubmit a previous solution unless that program was involved in an academic misconduct case. If misconduct was found, you must write a new version of that program.

We enforce this policy vigorously by running similarity detection software a few times per quarter over all submitted student programs, including programs from past quarters. Students who violate the policy are offered reduced scores and sometimes sent to a University committee. **This can lead to marks on permanent academic records**. Generally, several dozen students each quarter are given reduced scores for violating these policies. Please be careful, and please contact the instructor if you are unsure whether a particular behavior falls within our policy.