**CS& 141: Computer Science I (Java)**

Contents

[Course Details 1](#_Toc91059876)

[Prerequisites 1](#_Toc91059877)

[Course Outcomes 2](#_Toc91059878)

[Topic Coverage 2](#_Toc91059879)

[Required Materials 3](#_Toc91059880)

[Textbook 3](#_Toc91059881)

[Software 3](#_Toc91059882)

[Computer & Technical Requirements 3](#_Toc91059883)

[Grading and Assessment 3](#_Toc91059884)

[Purpose of Activities / Course Alignment 4](#_Toc91059885)

[Late Policy 4](#_Toc91059886)

[Academic Integrity 5](#_Toc91059887)

[Mandatory Decimal Grading 6](#_Toc91059888)

[Classroom Contract 6](#_Toc91059889)

[Discussion Expectations 7](#_Toc91059890)

[How to be Successful 7](#_Toc91059891)

[Getting Unstuck 8](#_Toc91059892)

[Letter of Recommendation Requests 9](#_Toc91059893)

[Student Services 10](#_Toc91059894)

[Campus Closures / Cancelled Class 10](#_Toc91059895)

[Access and Accommodations 10](#_Toc91059896)

[Tutoring Services 10](#_Toc91059897)

[Counseling Center 11](#_Toc91059898)

[Additional Campus Resources 11](#_Toc91059899)

[Privacy Policies 12](#_Toc91059900)

[Accessibility Policies 12](#_Toc91059901)

**Spring 2022 Quarter**

# **Course Details**

|  |  |
| --- | --- |
| **Instructor** | Crystal Hess - [chess@shoreline.edu](mailto:chess@shoreline.edu)  I prefer messages to be sent through Canvas.  Messages are checked daily and typical response time is within 24 hours, this excludes weekends and holidays. |
| **Office Hours** | Tues/Wed 9:30-11 AM and Sat 11-noon via zoom; or by appt. |
| **Class Hours** | Section #38671 - M/W 12:30pm - 2:30pm  Computer Lab 1308  Section #38668 - Online |

**CS& 141 - Computer Science I (5 credits)**

This course 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. Topics include: fundamental programming-in-the-small abilities and concepts, including procedural programming (methods, parameters, return, values), basic control structures (sequence, if/else, for loop, while loop), file processing, arrays, and an introduction to defining objects.

Due to the amount of material and quick pace for consumption, prior programming experience is expected.

## **Prerequisites**

ENGL& 101 (may be taken concurrently) and MATH& 141 or MATH 111 with grades of 2.0 or better. CS 121 or ENGR 202 with a grade of 2.0 or better.

## **Course Outcomes**

1. Develop solutions to small-scale problems, such as determining the best candidate for an admissions process, analyzing text for keywords, or playing a game of word guess.
2. Design and implement solutions to partially ambiguous problems, such as analyzing a set of data for patterns and information or building upon an existing code base.
3. Identify and use programming language constructs and coding patterns, such as iteration, methods, classes, arrays, fencepost problems, and token-based processing.
4. Break down problems into logical pieces using repetition, methods, and classes.
5. Test and debug solutions until they meet specifications.
6. Examine and compare reasoning, design, and effectiveness of alternative solutions.

## **Topic Coverage**

*Estimated* Topic Coverage

|  |  |  |
| --- | --- | --- |
| Week 1 | Chapter 1: Introduction to Java Chapter 2: Primitive Data and Definite Loops | Long Homework #0 |
| Week 2 | Chapter 2: Primitive Data and Definite Loops Chapter 3: Parameters and Objects | Long Homework #1 |
| Week 3 | Chapter 4: Conditional Execution | Exam #1 |
| Week 4 | Chapter 4: Conditional Execution Chapter 5: Program Logic and Indefinite Loops | Long Homework #2 |
| Week 5 | Chapter 5: Program Logic and Indefinite Loops |  |
| Week 6 | Chapter 6: File Processing | Exam #2 |
| Week 7 | Chapter 7: Arrays | Long Homework #3 |
| Week 8 | Chapter 8: Classes | Exam #3 |
| Week 9 | Chapter 9: Inheritance and Interfaces Chapter 10: ArrayLists |  |
| Week 10 | Chapters 1-10, ACM Graphics | Long Homework #4 |
| [Finals Week](https://www.shoreline.edu/apply-and-aid/registration/final-exams.aspx) | Wednesday, June 15 | Exam #4 |

# **Required Materials**

## **Textbook**

* Building Java Programs: A Back to Basics Approach, 5th Edition (older versions work) by Reges and Stepp (same book used for CS143)

## **Software**

* *[Java Development Kit (JDK)](http://www.oracle.com/technetwork/java/javase/downloads/index.html) (freely available online)*
* [*JGrasp IDE*](http://www.jgrasp.org/) *(freely available online)*

## **Computer & Technical Requirements**

* It is expected that when students enroll for this course they will be able to use a computer, be able to upload and download files, and successfully navigate browsers and websites.
* Students will receive homework throughout the course that must be completed on a computer with Internet access. The school library computers can be used for these purposes.
* **Need help?** Canvas questions can be directed to Shoreline e-Learning at [elearning@shoreline.edu](mailto:elearning@shoreline.edu), (206) 546-6966, visit Building 1200, or [24hour Canvas Support via chat or telephone](http://www.shoreline.edu/virtual-campus/elearning/canvas.aspx). Canvas and basic computing support is also available on campus at the [Shoreline Business Technology & eLearning Center](http://www.shoreline.edu/apply-and-aid/learning-support-centers/business-computer-software.aspx) in Building 1300, Room 1304.
* More [information on general technical requirements](http://www.shoreline.edu/virtual-campus/elearning/tech.aspx) can be found here.

# **Grading and Assessment**

In this course, you will be assessed through **Learning Practice** (daily practice assignments that are graded mostly for completion), **Individual Homework** (longer individual homework), **Exams** (assessing your ability to apply information within time bounds).

I strive to grade daily and weekly assignments within 5 days of the due date. The larger assignments take longer to grade and are generally graded within 7-10 days.

|  |  |
| --- | --- |
| **Activity** | **Possible Points** |
| Learning Practice | ~40 pts |
| Pair Programming | ~5 pts |
| Individual Homework | ~20 pts |
| Exams | ~40 pts |
| **TOTAL** | **~105 pts** |

## **Purpose of Activities / Course Alignment**

* **Learning Practice** (including Preview Quizzes, Practice Problems, Practice Reflections) gives you the opportunity to practice reading, writing, and reflecting on small segments of code. These assignments are mostly graded for completion. (Course Objectives 3, 4, and 6)
* **Pair Programming** creates a space for you to use vocabulary, negotiate use of different programming constructs, and test/debug with another person. These assignments must be completed with a partner. (Course Objectives 1, 2, 3, 4, and 5)
* **Long Homework** builds in difficulty throughout the course, providing an opportunity to build software solutions to tackle varying problems. (Course Objectives 1, 2, 3, 4, and 5)
* **Exams** provide an assessment of your understanding of programming vocabulary and ability to read and write code quickly. (Course Objectives 3 and 4)

## **Late Policy**

* Students are expected to turn in all assignments by the due date and time.
* Assignments not submitted on-time may receive zero points.
* Alternate arrangements *may* be approved for flexible due dates in special circumstances when the instructor is contacted **prior** to the due date. *No day-of extensions will be given. No assignments will be accepted beyond one week past an assignment’s due date.*

More information available here: [Shoreline's full Grades Policy (#6260)](https://www.shoreline.edu/about-shoreline/policies-procedures/documents/6260GradesPolicy.pdf)

## **Academic Integrity**

Academic honesty and integrity are expected. Shoreline Community College has a strict policy on cheating and plagiarism. You are cheating if you copy, steal, borrow, consult, or use any means to obtain answers from classmates or unauthorized sources for any assignment inside or outside class. Any student found guilty of cheating and/or plagiarism will receive a zero for the assignment. If it happens a second time, you may fail the course.

**Long Homework Assignments are individual.** All code you submit must be your own work. You may discuss general ideas of how to approach an assignment, but never specific details about the code to write. Any help you receive from or provide to classmates should be limited and should never involve details of how to code a solution.

* You **may not** work as a partner with another student on an assignment.
* You **may not** show another student your solution to an assignment for any reason.
* You **may not** look at another's solution for any reason.
* You **may not** have another person "walk you through" an assignment, describe in detail how to solve it, or sit with you as you write it. You also may not provide such help to another student. This includes tutors and current or former students.
* You **may not** post your homework solution code online to ask others for help. This includes public message boards, forums, file sharing sites and services, or any other online system.

Please note that **both parties involved in a cheating case are equally guilty**. That is, helping someone else cheat is just as bad as reading someone else’s solution. There are resources for students who are struggling; the right thing to do if you are approached for excessive “help” is to point the person to appropriate resources.

## **Mandatory Decimal Grading**

**Your grade will be calculated as the cumulative points that you earn over the quarter.** Expect that at the start of the quarter your grade will appear low as you build up points. This also means that your grade (total points and final GPA) will *never* go down.

**Final GPA decimal grading** will be used in accordance with the following table. There is not an option for pass/fail grading in this course as it is transfer-equivalent.

****

# **Classroom Contract**

You are expected to uphold the classroom contract as follows:

* **Be Prepared**: You are expected to complete all pre-work and homework before class starts so that you are ready to be engaged in classroom learning.
* **Be Present**: You are expected to be in class, on time, and actively present in the learning process.
* **Participate**: You are expected to be an active contributor in class. This means both asking questions and helping others.
* **Create space for learning:** You are expected to create and participate in a space where learning is conducive for all learners. This means that you participate in the class without dominating the learning space.

## **Discussion Expectations**

(Adapted from Dr. Susan Shaw, Oregon State University & Angela Velez-Solic, Indiana University Northwest)

* Commit to learning about, understanding, and supporting your peers.
* Assume the best of others in the class and expect the best from them.
* Recognize and value the experiences, abilities, and knowledge each person brings to class.
* Participate actively in the discussions.
* Think through and re-read your comments before you post them.
* Never make derogatory comments toward another person in the class. Do not demean or embarrass others.
* Do not make sexist, racist, homophobic, or victim-blaming comments at all.
* It is ok to disagree with ideas, but do not make personal attacks.
* Be open to being challenged or confronted on your ideas or prejudices.
* Challenge others with the intent of facilitating growth.
* Encourage others to develop and share their ideas.
* Be willing to change.
* Any form of hatred is considered serious and inappropriate.

## **How to be Successful**

* **Come to class**: Participating in class lectures and practice problems will give you structured time to absorb new concepts, practice them, and ask questions to the Instructor and your peers.
* **Ask for help early**: In programming, everything stacks, meaning what you learn today, you WILL use tomorrow. Therefore, if you start to fall behind, it is imperative that you ask for help as soon as possible.
* **Practice**: It is not enough in programming to just read about or listen to lectures about concepts. It is important to solve problems both in-class and on your own. You should work through practice problems (like on PracticeIt) until you are able to solve them without the aid of another person.
* **Utilize the textbook, slides, and videos provided**: While lecture will cover most of the concepts, the textbook, slides, and videos provided will provide more detail and reference outside of class.

## **Getting Unstuck**

Getting stuck is part of the programming process. However, one of the most useful skills you may learn from this course is "how to get unstuck."

Here are some suggestions:

* **Make systematic changes.** Look carefully at the work you have just done. Try changing one thing and see what effect it has.
* **Talk to a rubber duck.** No really, this is actually [a thing](https://en.wikipedia.org/wiki/Rubber_duck_debugging). Sometimes if you take a moment to tell someone (even someone non-techie) what you're trying to do, you'll end up having an Ah-Ha moment of clarity!
* **Ask a classmate.** Try to share ideas about how to figure out the problem rather than telling your classmate the answer. You'll learn as much by helping others find their mistakes as you will by finding your own.
* **Take a break.** Sometimes you just need to take a break! Looking at code with a fresh set of eyes solves problems quicker than beating your head against the computer :)
* **Ask the Instructor.** I'm here to help you. It's literally my job.

## **Letter of Recommendation Requests**

Letters of recommendation are often needed for applications to transfer universities, jobs, and internships. Instructors take pride in the letters that they write for students and can only craft strong letters for students whom them know well. Part of your job as a college student is to become the kind of student that a professor can speak highly of in a letter of recommendation — hardworking, capable, and intellectually inquisitive.

Before requesting a letter of recommendation, ask yourself:

* Have I discussed my academic or career goals with this instructor?
* Have I demonstrated an excellent work ethic or produced quality work in this course?
* Have I shown or communicated a passion for the subject or concepts that were taught?
* Will this instructor be able to incorporate personal and specific details about my academic growth or trajectory?

It is not necessary that the person writing your recommendation be able to speak to every bullet point above, but they should be able to speak to at least one of them.

Remember that requests may be declined for a variety of reasons, including, but not limited to: insufficient time to write the letter, not knowing the student well enough to provide specific character observations, or too much time has elapsed since working with the student.

**Requests for recommendations should come well in advance of the deadline for submission (i.e., at least 14 days in advance of the deadline, but the more notice the better).**

# **Student Services**

## **Campus Closures / Cancelled Class**

There are two types of suspended operations possible: campus is closed or classes are cancelled.  In the event of campus closure or cancelled class, students will be notified via Canvas of their expectations in relation to school work.

You can read Shoreline’s [Suspended Operations Procedures](https://www.shoreline.edu/about-shoreline/policies-procedures/documents/6030SuspendedOperationsProcedure.pdf)

## **Access and Accommodations**

Shoreline Community College is committed to providing educational programs without regard to disabling conditions as defined by Section 504 of the Rehabilitation Act of 1973. Reasonable accommodations will be made and no otherwise qualified individual with disabling conditions shall, on the basis of disability, be excluded from participation in, be denied the benefits of, or otherwise be subjected to discrimination under any program, activity or service administered by the college.

For more information: [Shoreline Student Accessibility Services](http://www.shoreline.edu/oss/students-with-disabilities/default.aspx)

## **Tutoring Services**

Tutoring Services provides students with free one-on-one tutoring support for any Shoreline Community College classes. Students can receive 2 hours of free tutoring a week per class they are registered for in a quarter.

They also have drop-in learning centers, such as the Computer Science and Engineering Learning Center, Biology/Chemistry Learning Center, Business Technology & eLearning Center, Physics Learning Center, Conversation Groups, and more (schedules and availability vary).

For more information and to apply for tutoring assistance or to apply to become a tutor, please visit our office in 4228 (Library), email [pttutors@shoreline.edu](mailto:pttutors@shoreline.edu), call 206-546-4776, or check out the webpage: [www.shoreline.edu/tutoring](http://www.shoreline.edu/tutoring)

## **Counseling Center**

The Counseling Center provides free, confidential and professional counseling services, resources, and referral to support the academic and personal success, health, and well-being of our students and campus community.

Students often visit the Counseling Center to discuss a wide variety of topics: depression, anxiety, relationship concerns, and stress management; indecision about major or career path; and academic concerns such as failing grades, struggling with a subject, or managing a learning disability. The Center also supports students who may be feeling suicidal or in crisis.

* FOSS – 5245, 206-546-4594, [www.shoreline.edu/counseling-center](http://www.shoreline.edu/counseling-center)

Need support when they are not available? For 24/7 emergency counseling, referral, or assistance please contact:

* King County: 24-Hour Crisis Line | 866-427-4747
* Snohomish County: 24-Hour Crisis Line | 800-584-3578
* Live Chat: [crisischat.org](http://crisischat.org/)
* Crisis Text Line: Text 741741
* 911 (for immediate health-related emergency)

## **Additional Campus Resources**

Check out the [Current Students page](https://www.shoreline.edu/currentstudents/) for more information about Academic Support, Student Services, Campus Life, and much more.

# **Privacy Policies**

To learn about the practices regarding personal information that may be collected from users in this course, check the privacy policies below:

* [PracticeIt](https://practiceit.cs.washington.edu/privacy-policy) Privacy Policy
* [Canvas](https://www.canvaslms.com/policies/privacy)
* [Panopto](https://www.panopto.com/privacy/)
* [Shoreline Privacy Policies](https://www.shoreline.edu/currentstudents/privacyandnondiscrimination.aspx)
* [Student Policies](https://www.shoreline.edu/currentstudents/student-policies.aspx)
* [Microsoft 365](http://www.microsoft.com/online/legal/v2/?docid=22&langid=en-us)
* [Google Privacy Policy](https://www.google.com/intl/en/policies/privacy/)

# **Accessibility Policies**

Below are links to accessibility policies for sites used within this course:

* [Canvas](https://www.canvaslms.com/accessibility)
* [Panopto Accessibility Features](https://support.panopto.com/articles/Documentation/accessibility-features)
* [Shoreline Community College](http://intranet.shoreline.edu/policies/documents/5000/5114.pdf) [(Audio Version)](http://intranet.shoreline.edu/policies/documents/5000/5114.pdf)
* [Microsoft Office 365](https://support.office.com/en-US/article/Accessibility-in-Office-365-ACA7ACCF-58A0-4467-BE5C-24A7E7933A9D?ui=en-US&rs=en-US&ad=US)
* [Google (covers YouTube)](https://www.google.com/accessibility/products-features.html)