

# **CSS 342: Data Structures, Algorithms, And Discrete Mathematics I**

## **Syllabus**

### **Course Information**

This course is designed as a fast-paced course along with CSS 343, in order to guide you for more advanced level CSS courses. It integrates the fundamental mathematics of computing with detailed instruction in program design. At the end of this course, you will be familiar with much of the C++ language and the basics of object-oriented programming. You will understand how to analyze a problem and design a solution. You will learn many basic data structures, algorithms, and the tradeoffs among memory, running time, and implementation time associated with them. Topics include: recursion, computational complexity and algorithm analysis, logic, mathematical proofs and induction, lists, stacks, queues, sorting and searching, data abstraction, and object-oriented methods.

### **Course Objectives:**

The objectives of this course are for students to:

- have a basic understanding of the C++ language and the basics of object-oriented programming.
- understand how to analyze a problem and design a solution
- know basic data structures, algorithms, and the tradeoffs among memory, running time, and implementation time associated with them.

### **Course Learning Outcome:**

Upon successful completion of the course, students shall be able to:

- CO-1 : Be able to perform conversion between number bases(a)
- CO-2 : Be able to implement computer programs to sort data(a)(i)

CO-3 : Be able to design and implement computer programs to solve problems using recursion(c)(i)

CO-4 : Be able to design and implement computer programs to solve problems using linked lists(c)(i)

CO-5 : Be able to formulate statements using proposition logic and compute their truth tables(e)

CO-6 : Be able to formulate induction proofs(e)

CO-7 : Be able to implement computer programs for lists, stacks, and queues(i)(k)

CO-8 : Be able to analyze a computer algorithm and determine its computational complexity(k)

### **Relationship of Course to Student Outcomes:**

This course supports the achievement of the CE student outcome:

- (a) An ability to apply knowledge of mathematics, science, and engineering.
- (c) An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environment, social, political, ethical, health and safety, manufacturability, and sustainability.
- (e) An ability to identify, formulate, and solve engineering problems.
- (k) An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.
- (i) A recognition of the need for, and an ability to engage in life-long learning.

### **Prerequisites**

- Calculus, Probability & Statistics, and two quarters of programming (C/C++ competency)
- Minimum grade of 2.5 in CSS 143; B CUSP 124

### **Text Book (Required)**

- *Data Abstraction & Problem Solving With C++: Walls & Mirrors*, Seventh Edition, Frank M. Carrano, Addison Wesley, 2017, ISBN-13: 9780134463971
- [An Active Introduction to Discrete Mathematics and Algorithms](#) by Cusack, 2018

### Grading Policy

Course Work	%
Midterm exam	25
Final exam	25
Quizzes	10
In-Class Work	10
Assignments	30
Total	100

Expected GPA grades. All grade change requests will be ignored.

Percentage	Grade
90 – 100	3.5 – 4.0
80 - 89	2.5 - 3.4
70 - 79	1.5 - 2.4
60 - 69	0.7 - 1.4

### Class Policies

**All the materials are either in class, or online. In-class only materials are unavailable unless you come to the class. The only exceptions are: you are terribly sick, family emergency, or have to go for an academic event. Even this case, you should provide evidence (doctor's note, for example).**

- **Assignments:** All assignments should be done independently and any cheating or collaboration work will be returned without points, and will be reported as an academic misconduct. All assignments are required to submit online (Canvas), unless otherwise noted. Please note that late assignments **will not be accepted** in any circumstances.
- **Exercise and Lab:** Short programming works or short questions will be given in class. You are required to finish the work in class and submit online (most cases). The works won't be repeated outside of the class, therefore it is your responsibility to be in class to

claim your points. Each will get a unique serial number for each exercise, and if do not provide the serial number, it won't be graded.

- **Exams:** You are given midterm and final exams. Midterm exam covers the contents you have learned in class by the time of the exam. Final exam will cover the topic after midterm exam. **There will be no make-up exam, either earlier or later. No exception.**
- **Quizzes:** will be given for the first 15 minutes of the class. 3 or 4 quizzes will be given in a quarter.
- **Special needs:** To request academic accommodations due to a disability, please contact [DRS \(Links to an external site.\)](#)[Links to an external site.](#). If you have a documented disability on file with the DRS office, please have your DRS counselor contact me and we can discuss accommodations. However, registration to DRS should be done **within 2 weeks after the class starts.**

## Lectures

The purpose of lecture is to explain concepts that are not completely clear from the book and work through practical examples. The lecture should not be the first time you've seen the material. My expectation is that you would have read the textbooks before coming to class as the lecture slides cannot cover everything the way you would understand fully.

## Communication

Any course related emails must begin with [CSS 342].

Before asking for help with any of your code, ensure you have done the following:

- Run the code through a static analyzer such as [CPPCheck. \(Links to an external site.\)](#)[Links to an external site.](#)
- Instrumented your code with output statements so that you know exactly where the problem is happening and what the value of any relevant variables are.

## Programs

Turn in a soft copy of your programming assignments on Canvas, **do not zip the files.**

**Each soft copy** must include your source code (in ASCII texts) , and, if specified, your report (in PDF or MS Word). Please check each homework specification about what you have to write in your report.

Your code should compile and run properly with **g++ compiler on Linux**, the professors will test your code only with this execution environment. Linux environment will be set up in the first lab.

Don't email your soft copy to the professors.

**Program grading guide:** Syntax errors and run-time errors without much output yield a low grade. Run-time errors (occurring after the majority of the output), or incorrect answers will result in a significant number of points being deducted from your grade. Otherwise, you will be graded on documentation (clarity and completeness), style (indentation, use of blank lines and spaces), meaningful identifier names, organization of your program (modularity, design), efficiency (no useless, unnecessary, or unnecessarily complicated code), output (clarity and format), the overall readability of your entire program, and following directions.

## Sample code

You can find some sample programs regarding standard template libraries, templates, recursion, sorting, lists, stacks, and queues in "Files-->examples" folder. You may refer to those programs when working on your lab work and homework. Note that you cannot read all files under this directory, since those unreadable files are key answers.

**URGENT CARE:** For anybody needing urgent support, please call *The Suicide Prevention Hotline 1.800.273.8258* or connect with the UWB CARE Team  
<https://www.uwb.edu/studentaffairs/care-team> Links to an external site.

## Disability

**Access and Accommodations:** Your experience in this class is important to us, and it is the policy and practice of the University of Washington to create inclusive and accessible learning environments consistent with federal and state law. If you experience barriers based on disability, please seek a meeting with Disability Resources for Students (DRS) to discuss and address them. If you have already established accommodations with DRS, please communicate

your approved accommodations to your instructor at your earliest convenience so we can discuss your needs in this course.

DRS offers resources and coordinates reasonable accommodations for students with disabilities. Reasonable accommodations are established through an interactive process between you, your instructor(s) and DRS. If you have not yet established services through DRS, but have a temporary or permanent disability that requires accommodations (this can include but not limited to; mental health, attention-related, learning, vision, hearing, physical or health impacts), you are welcome to contact DRS at 425.352.5307 or [rosal@uw.edu](mailto:rosal@uw.edu).

## **For Our Veterans**

Welcome! We at UW Bothell understand that the transition into civilian life can be challenging for our veteran students and we MANY resources for any who may want to reach out for guidance or assistance in these matters including our Vet Corp Navigator through the WDVA and our Student Veterans Association (SVA) and tutors/mentors for veterans. Please contact Veteran Services at 425.352.5307 or [rosal@uw.edu](mailto:rosal@uw.edu).

## **Academic Dishonesty**

Plagiarism and cheating are serious offenses and may be punished by failure on exam, paper or project; failure in course; and or expulsion from the University. For more information refer to the "[Policy on Academic and Behavioral Conduct \(Links to an external site.\)](#)[Links to an external site.](#)" policy in the University Catalog. For this class, it is permissible to assist classmates in general discussions of computing techniques. Each person, however, must develop his or her own solutions to the assignments and tests.

## **Student Misbehavior**

This class has zero-tolerance on student misbehaviors, such as disruptive talking, chronic avoidance of work, clowning, interfering with teaching activities, harassing classmates, verbal insults, rudeness to teacher, defiance, and hostility. Student misbehaviors will be reported to the school dean, as well as to the office of student conduct.

## **Remark**

- This syllabus represents a general plan for the course and deviations from this plan may be necessary during the duration of the course.
- Your constructive assessment of this course plays an indispensable role in shaping education at Washington State. When you are asked to do, please take time to fill out the course evaluation.

Please check [academic calendar \(Links to an external site.\)](#)[Links to an external site.](#) for any important dates, including registration, tuition fee deadline, withdrawal deadline and holidays.