

CSS 342 Mathematical Principles of Computing

Winter 2013

MW 5:45pm UW1-110

<http://courses.washington.edu/css342/timots/>

Instructor: Timothy Stewart (timots@u.washington.edu)

Office: UW1-349

Office Hours: M, W 7:45-8:30 pm; or by appointment

Course description:

This sequenced course integrates mathematical principles with detailed instruction in computer programming. Topics include presentation of formal arguments to prove mathematical statements; development of algorithms; sorting and searching; algorithm analysis; object-oriented programming; basic abstract data types including lists, stacks, queues, and binary trees.

Prerequisites: Calculus, two quarters programming

Grading:

Assignments 35% / Midterm exam 30% / Final exam 35%

- A scale of 90s (3.5-4.0), 80s (2.7-3.4), 70s (1.8-2.4), 60s (0.7-1.7) is a guide although not rigidly followed.
- Assignments consist of problems and programs, most notably in the assigned labs.

Texts:

Required textbooks:

- Data Abstraction & Problem Solving with C++ 5th ed, Frank M. Carrano, Addison-Wesley.
- Discrete Mathematics and Its Applications, 7th ed, Kenneth Rosen, McGraw-Hill.

Some other C++ books:

- C++ How to Program, , H.M. Deitel and P. J. Deitel, Addison-Wesley.
- The C++ Programming Language, Third edition, Bjarne Stroustrup, Addison-Wesley.
- C++ In Plain English, Brian Overland, Wiley.

C++ references:

- Effective C++ Second edition: 50 Specific Ways to Improve Your Programs and Designs;
- More Effective C++: 35 New Ways to Improve Your Programs & Designs, Scott Meyers, Addison-Wesley.
- Thinking in C++, Bruce Eckel, Prentice Hall.

Policies / Information:

- Work is to be done independently, collaboration of work is NOT acceptable, unless directed otherwise. You may discuss the problem statement with each other and help debug, but any actual work to be turned in must be done without collaboration. In other words, designing and coding is to be done independently. This class is run by honor code. By taking this class, you agree that you will not collaborate inappropriately

on any work. I know that in some cultures, family relationships and their loyalty are considered above all others. In this course, we are an academic family and you betray my and the university's trust should you violate the honor code. This violation will be taken seriously. For the Student Conduct Code, see http://www.uwb.edu/academic/policies/Academic_Conduct.xhtml.

- Computer use during lecture is limited to taking notes ONLY. You are not allowed to display any images on your screen during lecture as it is distracting to your neighbors trying to learn. Please turn off cell phones; any form of cell phone use during class time is prohibited.
- Assignments are due at Midnight on the posted due date. Late assignments are not accepted. No make-up exams will be given except under exceptional circumstances.
- To request academic accommodations due to a disability, please contact Disabled Resources (DRS) at 425.352.5307, 425.352.5303 TDD, 425.352.5455 FAX, or at dss@uwb.edu. You will need to provide documentation of your disability as part of the review process prior to receiving any accommodations.

Course goals:

The overall goal of CSS 342 is to learn discrete mathematics concepts and computer programming. You solve mathematical problems, present formal mathematical arguments, and program solutions to problems. You review searching and sorting algorithms, object-oriented programming, basic abstract data types, and study algorithm analysis. The C++ programming language is studied. Good software engineering techniques are encouraged throughout. As with most technical courses, besides ability and motivation, it takes time and a lot of hard work to learn and master the subject. Expect to spend an additional 15 hours a week outside of class time on average.

Assignments:

- Follow any special directions for turn-in given on an assignment. Always turn in an electronic copy of your source code (only .h and .cpp files) and other files asked for through Canvas.
- Your code should compile and run properly using the Linux g++ compiler.
- Coding and documentation style guidelines can be found linked off the assignments page at <http://courses.washington.edu/css343/zander/style.html>
- Syntax errors and run-time errors with little or no output yield a low grade. Run-time errors or incorrect answers will result in a significant number of points being deducted from your grade. In other words, CHECK YOUR ANSWERS!!!!!! Otherwise, you will be graded on documentation (clarity and completeness), style (indentation and use of blank lines/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, and following directions. A detailed grading rubric can be found at <http://courses.washington.edu/css342/timots/gradingRubric.pdf>.

Topics covered and tentative 342 schedule:

This is an approximate ordering of topics. Material will take about the allotted time and not all sections in all chapters are covered. Topics are labeled C++ or Math and

correspond to the appropriate textbook. Labs are proposed near the date listed and are due as indicated in Canvas.

Date	Topic	Material	Lab
01/07/13	Introduction, Preliminaries, etc.; C++ language structure, C++ structs, arrays (notes 1)	Website Course Notes; C++ Appendix A	
01/09/13	Software Engineering Principles; Searching, simple sorting (selection, bubble, insertion) (notes 2)	C++ 1; Website Course Notes; C++ 9.2 (some of)	Lab 0
01/14/13	Data Abstraction (Objects and Classes); C++ classes (notes 3)	Website Course Notes; C++ 3	
01/16/13	more C++ classes (notes 4)		Lab 1
01/21/13	Holiday – Martin Luther King Day		
01/23/13	Pointers & Linked lists (notes 5)	C++ 4.1-4.4	
01/28/13	more Pointers & Linked lists (notes 6)		Lab 2
01/30/13	Templates, STL intro (notes 7)	C++ 4.5, Appendix E	
02/04/13	Algorithm analysis (notes 8)	C++ 9.1; Math 2.4, 3.1-3.3	
02/06/13	Recursion (notes 9)	C++ 2; Math 5.3-5.4	Lab 3
02/11/13	Midterm exam , in class		
02/13/13	Sorting (using recursion) (notes 10)	C++ 9.2 (some of)	
02/18/13	Holiday - President's Day		
02/20/13	Binary Search Tree intro (notes 11)	C++ 10.3; Math 11.1-11.3	Lab 4

Date	Topic	Material	Lab
02/24/13	LAST DAY TO DROP A COURSE		
02/25/13	Object-oriented design; Stacks (<u>notes 12</u>)	C++ 1, C++ 6	
02/27/13	Queues (<u>notes 13</u>)	C++ 7	Lab 5
03/04/13	Recurrence and Induction (<u>notes 14</u>)	C++ 2.1, 5.3; Math 5.1-5.2	
03/06/13	more Recurrence and Induction (<u>notes 15</u>)	C++ Appendix D; Math 8.1, 8.3	
03/11/13	Representation of Integers; Mathematical Foundations (<u>notes 16</u>)	Math 4.2; Math 1.1, 1.3-1.5	Lab 6
03/13/13	Intro to Design Patterns, last day stuff (<u>notes 17</u>)	Website Course Notes	
03/18/13	Final exam , in class		

Session notes may be found at:

notes 1: [http://courses.washington.edu/css342/timots/Notes/session 1 web.pdf](http://courses.washington.edu/css342/timots/Notes/session%201%20web.pdf)

notes 2: [http://courses.washington.edu/css342/timots/Notes/session 2 web.pdf](http://courses.washington.edu/css342/timots/Notes/session%202%20web.pdf)

etc.