Computer Science II

Professor: Will Crissey, Jr.

Section 001   Room: College of Arts and Sciences, 142. Lab: CAS 241
Class: Mo, We, and Fr   3:20 PM - 4:10 PM, Class number 77337

Office: CAS 239

Phone: 330-972-8013

Email: wcrisseyjr@uakron.edu

Office Hours: http://www.cs.uakron.edu/~wcrisseyjr/Fall%20schedule%202017.htm

Other times are available and may be arranged by appointment.

Instructor Web Page: http://www.cs.uakron.edu/~wcrisseyjr

Prerequisite: Completion of 3450:208 (Introduction to Discrete Mathematics) and 3460:209 (Computer Science I) with C- or better or equivalent.

Course Description:

- **Rationale**
  To write effective computer code, one must employ techniques like analyzing algorithms, deploying these algorithms as programs, and implementing, debugging and testing the resulting programs. The goal of the 209-210 sequence is to introduce these techniques and lay the foundation for more advanced courses in computer science and engineering.
- **Objectives**
  - To understand the fundamentals of basic computer organization, efficient problem solving and structured computer programming.
  - To apply and demonstrate mastery of the presented course concepts by designing and coding a series of programs to solve real-world problems using the C++ programming language and either the Linux or MS Windows o.s.
  - To exhibit the ability to teach and learn from others.

Text Book:

Savitch, Problem Solving with C++, 9th Edition (9th Ed.) (Pearson, 2015), ISBN-10: 0-13-386222-4, ISBN-13: 978-0-13-386222-5 (may differ: make sure it is 9th edition).

Labs:

The lab is the Computer Science Department run hands on lab. All of that lab work will be done using the University's Lab in the College of Arts and Sciences, room 241. Check your schedules and check the due dates for the labs. The CS labs are designed to challenge you, and offer different learning opportunities and objectives than normal book learning. You will also gain exposure to the LINUX environment, and learn how to navigate and program on this open source platform.

Grading: The course grade is based on a student's overall performance through the entire semester. The final grade is distributed among the following components:

- Lab: 10%

- Assignments: 10%

- Projects: 25%

- Midterm Exams: 30%

- Final Exam: 20%

- In Class: 5%

Lab (10%)

There is a Computer Science Department run lab, which takes place each week throughout the semester on your assigned lab day. The lab attendance is not required for these sessions, but highly encouraged. Weekly labs are to be submitted by the Friday at 11:59 PM during the week that they are assigned (10%).

Through interactive and hands on practice, these labs will provide a means to apply the material projected in class and outside of the normal content.

Assignments (10%)

Programming assignments will be given throughout the semester and are coordinated in a parallel relationship with the current topics presented in class. These programs are a great way to apply what you have learned.

Projects (25%)

An individual, continuing project will be assigned during the semester. This will strongly correlate to the material and integrate the skills learned with the practice of programming both design and solution. The details will be available as the parts of the project are assigned.

Midterm Exams (30%)

There will be two midterm tests. A comprehensive view of the materials will be shared. The tests will be scheduled.

Final Exam (20%)

The final exam will be at the close of the semester, usually during exam week. The final is similar to the midterm and a comprehensive view of the materials will be shared.

In Class (5%)

The "in class" category involves participation in events during the regular class times. Each and every activity takes place at arbitrary times and is aligned with the topics, and so you really want to be in attendance so that you can participate and receive points. You cannot submit these assignments without being in attendance. Attendance has been proven to be critical for success in this class. The information and activities during class stimulate thinking and reasoning by acquiring and applying knowledge. Activities also provide immediate practice of the material presented in class. Please note that certain materials that are covered only in class may be part of the exams.

Grading scale:
A: 93%-100% ; A-: 90%-92.99% ; B+: 87%-89.99% ; B: 83%-86.99%; B-: 80%-82.99%; C+: 77%-79.99%; C: 73%-76.99%; C-: 70%-72.99%; D+: 67%-69.99%; D: 63%-66.99%; D-: 60%-62.99%; F: Less than 60%. Final grades are *not* subject to rounding up.

Course Outline: This course is divided into three units. Springboard is the main driver of course content and the organization reflects the units. Class lectures may vary from online to in class. Each unit builds upon the material from the book, and has

prerequisites of timelines before allowing one to proceed onward with viewing subsequent materials, etc... Assignments, for example, cannot be started and are not viewable until the material begins. In Springboard a student is expected to go through each unit's materials and self-certify by passing any quizzes and practice exams.

Examination Policy: All exams include material covered in the classroom, assignments and class exercises before the exam date. A number of study materials, including mock exams (quizzes), will be used for each test and will be posted and discussed in advance. **The tests will be closed book, closed notes.**

Make-Up Exam Policy: Ordinarily, no make-up examinations will be given. Make-up examination will be considered only in case of documented emergencies and only when the instructor is notified about the emergency as soon as possible. If this is not done, then the grade is automatically a 0 for that examination. It is the responsibility of the student to contact the instructor for arranging a make-up. Written verification for the student's inability to take an exam will be required and must be approved by the university's standard process for student emergencies.

Homework Policy: All homework assignments and projects are due on the specified date. An assignment must be in the correct location in the repository or Springboard by or before 11:59 PM on the specified day. LATE ASSIGNMENTS ARE NOT ACCEPTED. All assignments must be individually and independently completed and must represent the effort of the student turning in the assignment. When asking for help, you may have your code viewed by others, but you may not view theirs. Students should be aware of and avoid plagiarism. IMPORTANT: Plagiarism will result in a zero grade for the entire category in the grade book. Also, should two or more students turn in noticeably the same solution or program, in the judgment of the instructor, the solution will be considered a group effort and the result the same. Anyone involved in either plagiarized or group-effort effort will be given a zero grade for the category (lab/project/assignment) and a zero grade for all submissions (retroactively to the first submission). A student turning in a group-effort or plagiarized homework/project/assignment more than once will automatically receive an "F" grade for the course. Submission of work that is entirely or partly not yours will be reported to the *Department of Student Conduct and Community Standards*.

Late Assignment/Project: An assignment is to be turned in on the day it is due. An assignment cannot be turned in late, and therefore will NOT be graded (0%). Please refer to The Programming Rubric. Three categories distribute points accordingly for the assignment. Projects: A project that is in the Subversion repository or Springboard before 11:59 PM on the specified day will be graded. Anything submitted after the due date will *not* be graded (0%). Please refer to the specific Project's Rubric and the categories for point distribution for the project assignment.

Registration: All students attending the class must register. Students whose names do not appear University's official 15-day class list will not be permitted to participate (attend class, take exams, or receive credit.) Note that the University has a new withdrawal policy. Consult University information for specific dates and policies.

Academic Honesty: All submitted work (assignments, projects, tests, etc...) must be your own. Students should be aware of and avoid plagiarism. Submission of work that is entirely or partly not yours will be reported to the *Department of Student Conduct and Community Standards*. (http://www.uakron.edu/sja/)

Code of Student Conduct: Refer to 3359-41-01, code of student conduct of the University of Akron. The standards of conduct and scholarship outlined by the university will be the policy in our classroom. Non-compliance or deviations, such as disruptive behavior, can impact class participation credit. Continued non-compliance can lead to further disciplinary actions.

Title-IX: The University of Akron is committed to providing an environment free of all forms of discrimination, including sexual violence and sexual harassment. This includes instances of attempted and/or completed sexual assault, domestic and dating violence, gender-based stalking, and sexual harassment. Additional information, resources, support and the University of Akron protocols for responding to sexual violence are available at uakron.edu/Title-IX

Special Notice: Any student who feels she/he may need an accommodation based on the impact of a disability should contact the Office of Accessibility at 330-972-7928. The office is located in Simmons Hall

*Note: This syllabus is subject to change based on the needs of the class. Topics and times are tentative.

Week / Beginning Date	Unit Module	Topic	Reading/labs, projects and assignments
Week 1 / Aug 28	UNIT 1 Module 1	Class intro, Review of Pointers, Structures	Reading: Chapter 9, Chapter 10, Section 10.1
Week 2 / Sept 4	UNIT 1 Module 2, 3	Structures continued, Classes	Reading: Chapter 10, Section 10.2 **No classes or labs Monday** the 4th Labor Day observance - University closed CS Lab 1 - CS Account setup and the shell Assignment A is due Wednesday @ 11:59 PM
Week 3 / Sept 11	UNIT 1 Module 4, 5	Classes continued, Arrays of Objects	Reading: Chapter 10, Section 10.3, Chapter 11, Section 11.3 and 11.4 CS Lab 2 - Building software with `cmake` Project Part A due Friday @ 11:59 PM
Week 4 / Sept 18	UNIT 1 Module 6, 7	Abstract Data Types, Friends, Operator Overloading	Reading: Chapter 11, Section 11.1 and 11.2 CS Lab 3 - Debugging Assignment B is due Wednesday @ 11:59 PM Assignment C is due Wednesday @ 11:59 PM
Week 5 / Sept 25	UNIT 1 Module 8	Operator Overloading continued	Reading: Chapter 11, Section 11.2 Assignment D due Wednesday @ 11:59 PM No labs this week!
		Midterm review	Review UNIT 1 – Modules 1 through 8
		Midterm	**MIDTERM EXAM I** – Midterm covers UNIT 1
Week 6 / Oct 2	UNIT 2 Module 1, 2	Namespace and Separate compilation, Templates	Reading: Chapter 12, Section 12.1 and 12.2, Chapter 17, Section 17.1 - 17.2 Project Part B due Friday @ 11:59 PM
Week 7 / Oct 9	UNIT 2 Module 3, 4	Linked Lists, Stacks	Reading: Chapter 13, Section 13.1 CS Lab 4 - Text Filters Assignment E is due Wednesday @ 11:59 PM
Week 8 / Oct 16	UNIT 2 Module 4, 5	Stacks cont., Queues, The Standard Template Library: Iterators	Reading: Chapter 13, Section 13.2 Chapter 18, Section 18.1 CS Lab 5 – Objects and Constructors Assignment F is due Wednesday @ 11:59 PM
Week 9 / Oct 23	UNIT 2 Module 5, 6	The STL: Containers	Reading: Chapter 18, Section 18.2 CS Lab 6 - Pretty pictures Assignment G is due Wednesday @ 11:59 PM Project Part C due Friday @ 11:59 PM

Week 10 / Oct 30	UNIT 2 Module 6	The STL: Containers cont.	Reading: Chapter 18, Section 18.2 No labs this week!
		Midterm review	Review UNIT 2 – Modules 1 through 7
		Midterm	**MIDTERM EXAM II** – Midterm covers UNIT 2
Week 11 / Nov 6	UNIT 3 Module 1, 2	The STL: Algorithms, Writing a Vector Class	Reading: Chapter 18, Section 18.3 CS Lab 7 - Range algorithms Assignment H is due Wednesday @ 11:59 PM
Week 12 / Nov 13	UNIT 3 Module 2, 3	Writing a Vector Class cont., Recursion	Reading: Chapter 14, Section 14.1, 14.2, 14.3 CS Lab 7 – Range algorithms continues Project Part D due Friday @ 11:59 PM
Week 13 / Nov 20	UNIT 3 Module 3, 4	Exceptions, Binary Trees	Reading: Chapter 16, , Section 16.1, 16.2 Chapter 13, Section 13.1 **No classes or labs Thursday** the 23rd, and **Friday** the 24th Thanksgiving Day observance - University closed CS Lab 8 – Generic algorithms
Week 14 / Nov 27	UNIT 3 Module 5	Binary Trees continued, special topic TBA	Reading: Lecture Notes CS Lab 9 – Exceptions Assignment I is due Wednesday @ 11:59 PM
Week 15 / Dec 4	UNIT 3 Module 6	TBA continued	Reading: TBA Assignment J is due Wednesday @ 11:59PM CS Lab 10 – Recursion (Final lab!)
		Final review	Review for final exam **FINAL EXAM** – Comprehensive
Week 16 / Dec 11		Finals week	Final Exam take home problem - due Sunday, Dec 10 @ 11 :59 PM Project Part E due Monday Dec 11 @ 11:59 PM

Time and dates may be subject to change

All assignments, labs and projects are due at 11:59 PM, unless noted otherwise
CS Labs are located in the College of Arts and Sciences, room # 241

Class activities matrix with all the dates...

WEEK	Topics	Assignment	Computer Sci Lab	Project	Exam
Week 1	Class intro, Review of CS1	None	None		
Week 2	Structures, Classes	Assignment 1	CS Lab 1 - CS Account setup and the shell		
Week 3	Arrays of Objects,	Assignment 2	CS Lab 2 - Building software with cmake		
Week 4	Abstract Data Types, Friends, Operator Overloading	Assignment 3	CS Lab 3 - Debugging		
Week 5	Abstract Data Types, Friends, Operator Overloading cont...	Assignment 4	No CS Labs this week!		**Mid-term 1**
Week 6	Namespace and Separate compilation, Templates	None	No labs this week – no class Tuesday		
Week 7	Linked Lists, Stacks	None	CS Lab 4 - Text Filters		
Week 8	Stacks cont., Queues, The Standard Template Library	Assignment 5	CS Lab 5 – Objects and Constructors		
Week 9	The Standard Template Library continued	Assignment 6	CS Lab 6 - Pretty pictures	Project 1	
Week 10	Midterm review Extended Lecture	Assignment 7	No CS Labs this week!		**Mid-term 2**
Week 11	Wrap up STL, Writing a Vector Class	None	CS Lab 7 - Range algorithms		
Week 12		None	CS Lab 8 – Generic algorithms		
Week 13	Recursion, Exceptions	Assignment 8	CS Lab 8 – Generic algorithms continues		
Week 14	Binary Trees	Assignment 9	CS Lab 9 – Exceptions		
Week 15	TBA Final review	Assignment 10	CS Lab 10 – Recursion		**Final**
Week16				Project 2 (on 5/8)	

Lab/Teaching Assistants

Your LAB and grader information: https://dev.cs.uakron.edu/trac/cs210fa17

ADDITIONAL RESOURCES

Other C++ Textbooks (and some links):

- Gaddis, *Starting Out with C++ from Control Structures through Objects* (8th Ed.) (Pearson, 2015).

- Dale and Weems, *Programming and Problem Solving with C++* (Jones and Bartlett, 2009). DW link.

- Deitel and Deitel, *C++: How to Program, 8th Ed.* (Prentice Hall, 2011). DD link.

- Bjarne Stroustrup (Creator of C++), "The C++ Programming Language", 4th ed, 2012. (Covers C++11.)

- Nicolai M Josuttis, "The C++ Standard Library: A Tutorial and Reference", 2nd ed, 2012.

- Bjarne Stroustrup, "The Design and Evolution of C++".

- Stephen Prata, "C++ Primer Plus Developer's Library", 6th ed, Addison-Wesley, 2012. (Covers C++11.)

- Stanley B. Lippman, Josee Lajoie, Barbara E Moo "C++ Primer", 5th ed, 2012. (Covers C++11.)

- Herb Sutter, "Exceptional C++: 47 Engineering Puzzles, Programming Problems and Solutions", 1999; "More Exceptional C++: 40 New Engineering Puzzles, Programming Problems and Solution", 2001.

- Gary J. Bronson, "Program Development and Design using C++".

- Robert C. Seacord, "Secure Coding in C and C++", 2nd ed, 2013.

Useful C++ Online References (including IDE and compiler links):

- http://www.cplusplus.com (C++ documents, tutorials, and references).
- http://geosoft.no/development/cppstyle.html
- Bjarne Stroustrup's C++11 FAQ @ http://www.stroustrup.com/C++11FAQ.html.
- GNU GCC (GNU Compiler Collection) @ http://gcc.gnu.org, with source codes.
- GNU GCC Manual @ http://gcc.gnu.org/onlinedocs.
- GNU 'make' manual @ http://www.gnu.org/software/make/manual/make.html.
- CodeBlocks Mother Site @ http://www.codeblocks.org/.
- CodeBlocks Wiki @ http://wiki.codeblocks.org/index.php?title=Main_Page.
  CodeBlocks User Manual @ http://www.codeblocks.org/user-manual.