Computer Science I

Professor: Will Crissey Renwald

Section 020   Room: Leigh Hall, 311. Lab CAS 241
Class: Mo, and We   10:15 AM - 11:30 AM, Class number 70506

Section 030   Room: Arts and Sciences, 407. Lab CAS 241
Class: Mo, and We   11:45 AM - 1:00 PM, Class number 70509

Office: CAS 239

Phone: 330-972-8013

Email: wcrisseyjr@uakron.edu

Office Hours: http://www.cs.uakron.edu/~wcrisseyjr/Fall%20schedule%202018.htm

Other times are available and may be arranged by appointment.

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

Prerequisite: 3450:145 (Algebra for Calculus) or 3450:149 (Precalculus Mathematics)

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++, 10th Edition (10th Ed.) (Pearson, 2018), ISBN-13: 9780134448282. (*Paper Bound*).

Savitch, Problem Solving with C++ Plus MyProgrammingLab with Pearson eText -- Access Card Package, 10th Edition (10th Ed.) (Pearson, 2018), ISBN-13: 9780134710747. (*Paper Bound with Access Card*).

Labs:

There are two parts to the lab. The first 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. The second lab is a remote lab that is offered by the publisher. The publisher's lab is called myProgrammingLab (MPL). It is an interactive software tool that offers many exercises that will enhance the learning process. You must purchase this lab along with the book. You will need an MPL access code that is bundled with the book.

Once you are set, you will need to go to:
http://www.pearsonmylabandmastering.com/northamerica/myprogramminglab/

- **Section 020 - MPLID 83043 use Section Access Code: UOFA-35079-MCHT-38**
- **Section 030 - MPLID 83045 use Section Access Code: UAAB-35081-AAVV-38**

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:

- Labs (hands on and remote): 20%

- Assignments: 5%

- Projects: 20%

- Midterm Exams: 30%

- Final Exam: 20%

- Participation: 5%

Labs (20%)

There are <u>two</u> labs for our class:

- The Computer Science Department run lab, which takes place each week throughout the semester on your assigned lab day. The lab attendance is required for these sessions. Important: you will not receive credit for any lab that you do not attend. Make sure you sign in. Weekly labs are to be submitted by the end of the lab session (10%)

- The publisher's lab, called MyProgrammingLab, or MPL. This interactive tool has some questions and a lot of good coding practice, and it can be done anytime and anywhere that you have access to the internet. It follows along with the book. There are due dates built right into the MPL tool. You can enter for late credit until through the following Sunday after the due date. MPL grades are calculated as follows: (#correct) plus (the #correct but late divided by 2). Due dates are Monday's, 11:59 p.m. (10%)

These labs provide a means to apply the material projected in class through interactive, hands on practice.

Assignments (5%)

Programming assignments will be given throughout the semester and are coordinated in a parallel relationship with the current topics presented in class. There are nine total assignments that are available, and some are optional (bonus). These programs are a great way to apply what you have learned. No bonus points for assignments are available to those missing 4 or more classes (see attendance), and no bonus assignments will be graded if you have not completed **all** the required assignments. So if you missed a regular one cannot get credit.

Bonus assignments are: Assignment 2, 4, 7, and 9. These are optional!

Projects (20%)

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.

Attendance (5%)

There is a minimum attendance policy and you must be in attendance to receive participation points. Attendance has been proven to be critical for success in this class. For a semester, more than 4 total missing activities, or absences, will result in a zero for participation. Also, no bonus points are available to those with a zero for participation. That means if you miss 4 classes, you are ineligible for any bonus points for this class ☹

Please further note that the instructor reserves the right to withdraw you from the class after accumulating more than 4 total absences. Also see code of conduct. 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. Brightspace is the main driver of course content and the organization reflects the units.

Examination Policy: All exams include material covered in the classroom, assignments and class exercises before the exam date. A number of study materials, including prac exams, 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 *before* the exam. 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 Brightspace by or before 11:59 PM on the specified day. LATE ASSIGNMENTS ARE NOT ACCEPTED. **IMPORTANT:** If you request a late submission to the professor or the TA without any legitimacy for such a request, you will receive a zero for the entire project or assignment category. If you have mechanical difficulties you must email the issue (must also include screen copies and error messages) before the due date's time ends, and immediately submit the late work as soon as the problem has been resolved.

Plagiarism: All work (projects, assignments, and exams) must be individually and independently completed and must represent the effort of the student turning in the assignment. When asking for help from other students, you may have your code viewed by others, but you may not view theirs. When asking for help anything or anyone outside of the class, understand that any code that is not that of the student's code is *not* considered to have been independently completed and will be construed as plagiaristic. 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 (in the rubric) distribute points accordingly for the assignment. <u>Projects:</u> A project that is in the Subversion repository or Brightspace 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. Do not share or copy code from any public domain. 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.

Class material: All class materials, including code, instructions, and any type of document are proprietary and confidential information of Professor Will Crissey, Jr., and is to be treated as Confidential Information. No information will be Confidential Information that: (i) is already known to Student, or (ii) is or becomes publicly known through no wrongful act of Student, or (iii) is received by Student from a third party without similar restrictions. Student will not disclose any Confidential Information to any other person. Student will not use any Confidential Information other than in connection with the class. Students who violate this confidentiality will be reported to be reported to the *Department of Student Conduct and Community Standards*, will receive a zero for this course, and will be liable for any expectation, restitution, and/or recovery damages.

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	Topic	Reading/labs, projects and assignments
Week 1 / Aug 27	UNIT 1	Introduction to Computers and C++ Programming, and C++ Basics	Reading: Chapter 1 and Chapter 2 No labs this week!
Week 2 / Sep 3	UNIT 1	C++ Basics continues	Reading: Chapter 2 CS Lab 1 - CS Account setup and using the Code::Blocks IDE **No classes or labs Monday** the 3rd Monday - Labor Day observance - University closed
Week 3 / Sep 10	UNIT 1	C++ Basics continues and More Flow of Control	Reading: Chapter 3 CS Lab 2 - Program Documentation and Standards **MPL** Chapter 1 due Monday @ 11:59 PM Assignment 1 due Monday @ 11:59 PM
Week 4 / Sep 17	UNIT 1	More Flow of Control continues	CS Lab 3 - Data Validation **MPL** Chapter 2 is due Monday @ 11:59 PM Assignment 2 is due Monday @ 11:59 PM
Week 5 / Sep 24	UNIT 1	Midterm review	Review Chapters 1 through 3 **MPL** Chapter 3 is due Monday @ 11:59 PM Assignment 3 is due Monday @ 11:59 PM **Project Part 01** due Friday @ 11:59 PM CS labs are exams this week. Wed/Thurs.
		Midterm	**MIDTERM EXAM I** – Midterm covers Chapter 1 through 3. Midterms are taken in the lab, Wednesday/Thursday
		Procedural Abstraction	Reading: Chapter 4 (4.1 Top-down design pp. 182-183 and 4.4 Procedural abstraction pp. 204-207)
Week 6 / Oct 1	UNIT 2	Procedural Abstraction and Functions That Return a Value	Reading: The rest of Chapter 4 No labs this week.

Week 7 / Oct 8	UNIT 2	Functions for All Subtasks	Reading: Chapter 5 CS Lab 4 - Functional Decomposition and Design **MPL** Chapter 4 is due Monday @ 11:59 PM Assignment 4 is due Monday @ 11:59
Week 8 / Oct 15	UNIT 2	I/O Streams as an Introduction to Objects and Classes	Reading: Chapter 6 CS Lab 5 - More on Design and Refactoring **MPL** Chapter 5 is due Monday @ 11:59 PM Assignment 5 is due Monday @ 11:59 PM
Week 9 / Oct 22	UNIT 2	Functions and I/O continues	Reading: Chapter 4, 5, 6 CS Lab 6 - Testing and Debugging
Week 10 / Oct 29	UNIT 2	Midterm review	Review Chapters 4 through 6 **MPL** Chapter 6 is due Monday @ 11:59 PM Assignment 6 is due Monday @ 11:59 PM **Project Part 02** due Friday @ 11:59 PM No CS Labs this week - Exams instead.
		Midterm	**MIDTERM EXAM II** – Midterm covers Chapter 4 through 6. Midterms are taken in the lab, Wednesday/Thursday
		Arrays and Searching and Sorting Arrays	Reading: Chapter 7 (7.1 Introduction to arrays pp. 378 – 386)
Week 11 / Nov 5	UNIT 3	Arrays and Searching and Sorting Arrays	Reading: The rest of Chapter 7 CS Lab 7 – More on Testing and Debugging
Week 12 / Nov 12	UNIT 3	Continue Arrays and Searching and Sorting Arrays Strings and Vectors	Reading: Chapter 8 CS Lab 8 – Programming Efficiency **MPL** Chapter 7 is due Monday @ 11:59 PM Assignment 7 is due Monday @ 11:59 PM
Week 13 / Nov 19	UNIT 3	Continue Strings and Vectors	Continue Chapter 8 CS Lab 9 – Error Handling
Week 14 / Nov 26	UNIT 3	Pointers	Reading: Chapter 9 **MPL** Chapter 8 is due Monday @ 11:59 PM Assignment 8 is due Monday @ 11:59 PM **No classes** or labs Thursday and Friday this week Thanksgiving observance - University closed
Week 15 / Dec 3	UNIT 3	Continue Pointers,	Continue Chapter 9 **MPL** Chapter 9 is due Monday @ 11:59 PM

		Finals review	Assignment 9 is due Monday @ 11:59 PM
		Final Exam	No Labs Tuesday and Thursday – go study!
			Review for final exam
			FINAL EXAM – Mainly on Chapters 7, 8, 9
			Exam date: Thursday/Friday in the lab
Week 16 / Dec 10		Final Project	**Project Part 03** due Wednesday @ 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

Lab/Teaching Assistants

- Section 020 (Tuesday, Thursday 1:10 - 2:00): Ms. Alekhya Choppadandi

    o Email: nc72@zips.uakron.edu>

    o Office Hours: (Thursday 7:00 - 9:00); CAS 254.

- Section 030 (Tuesday, Thursday 12:10 - 1:00): Mr. Brian Kovacs

    o Email: bck25@zips.uakron.edu

    o Office Hours: (Tuesday 5:00 - 7:00); CAS 254.

You may go to either of these office hours!

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.