

SYLLABUS

CSCI 1300: COMPUTER SCIENCE 1: STARTING COMPUTING

SPRING 2017, 4 CREDITS, JANUARY 17 – MAY 11, 2017

Class meeting times: Lecture Monday, Wednesday, Friday 1:00-1:50pm

Class meeting location: MATH 100

Recitations meeting times: vary

Recitations meeting location: ECCR 211

INSTRUCTOR INFORMATION

Name: David Knox

Email: david.knox@colorado.edu

Office Location: ECOT 732

Office Hours: Th at 1pm–3pm, Fri at 1pm–2pm, or by appointment

COURSE INFORMATION

Fit within curriculum: required foundation course for Computer Science BS and BA students.

Course description and prerequisites: Teaches techniques for writing computer programs in higher level programming languages to solve problems of interest in a range of application domains. This class is intended for students with little to no experience in computing or programming. Credit not granted for this class and CSCI 1310 or CSCI 1320 or ECEN 1310. Requisites: Requires a prerequisite or corequisite course of MATH 1300 or MATH 1310 or APPM 1345 or APPM 1350 (all minimum grade C-).

TEXTBOOKS AND MATERIALS

Required text: Problem Solving with C++, Ninth Edition. Savitch. Pearson Education. 2015. ISBN: 9780133591743.

Other supplemental materials: The Practice of Computing Using Python, Second Edition. Punch and Enbody. Pearson. 2012. ISBN: 9780132805575.

It is highly recommended that you get a Dropbox account, or invest in a USB memory stick, to save files created in the virtual machine environment. This environment is not backed up, and if it crashes, you will lose all of your work.

Course materials, such as lecture notes and assignments, will be available in electronic form on the Moodle site for the course: <http://moodle.cs.colorado.edu/>.

COURSE OUTCOMES

This class covers the basics of computer programming using C++. The focus will be on understanding the basic components of computer languages. Upon completion of the course, the student can:

- Design and construct algorithms for problem solving by applying processes of abstraction and program decomposition.
- Implement fundamental programming constructs, such as variables, expressions, conditionals, and iterative control structures, in a higher-level language.
- Evaluate and implement simple I/O, such as user input and file I/O, including the necessity for external input to a program and the role of external data storage.
- Design and explain the role of functions in program construction, including an understanding of parameter passing and return values.
- Describe the properties of data types, including the primitive types of numbers, characters, and booleans, as well as more complex data types, such as arrays, records, and strings.
- Use an Integrated Development Environment to produce code that is free of syntactical, logic, and run-time errors. Understand the process of debugging as part of software development.
- Design and create code using object-oriented design methodology, including an understanding objects and classes, information encapsulation, and efficient class design.
- Use third-party code to accomplish a programming objective. This includes learning to read code written by another individual and modifying the code, or using third-party libraries.
- Develop understanding that software development is a dynamic, social process, and that learning how to seek out information is a necessary skill for success.

Brief list of topics to be covered:

- Computer architecture and environment
- Variables and data types
- Variables and operators
- Functions
- Control structures: if statements
- Control structures: if/elif/else statements
- Control structures: while loops
- Control structures: for loops, iteration
- Strings: indexing, iteration, comparing
- File I/O
- Lists
- Dictionaries
- Classes and objects
- Classes, functions, methods
- Arrays
- I/O Streams

GRADING

Recitation	30%	(Weekly, due by 5pm on Sunday)
Homework	40%	(Weekly, due by 12:30pm on Friday)
Projects	10%	(One project)
Lab Practicum**	20%	(Four midterms, equally weighted)

** MUST receive a cumulative score of at least a 65 average on the practicums OR YOU CANNOT RECEIVE BETTER THAN D+ in the course.

ADDITIONAL LAB PRACTICUM GRADE REQUIREMENTS TO PASS THE CLASS

There are four midterms in this class that account for 20% of your final grade. There are 400 points possible on the midterms, which break down to 100 points per practicum. You must earn at least 260 points on the four midterms to receive better than a D+ in the class, regardless of your scores on the other aspects of the class. The practicums are an individual assessment of your ability to apply the programming skills learned in lectures, recitation assignments, and homework assignments. The 260 points is equivalent to a 65% (D) average on your midterms. If you do not show a skill level that is at or above our threshold, you will not receive a grade better than D+. A grade of a C- in this class is required to take the next class in the computer science sequence. The midterm schedule is given in the class schedule shown below.

The final exam timeslot can be used to take one or more replacement midterm practicum. If you do not earn the minimum exam points to pass the class, you can attend the final exam time period to take a replacement midterm. Even if you do earn the minimum midterm grade, you are still welcome to take a replacement midterm to bring up your grade. However, if you attempt the replacement practicum, the new score will replace your previous score, even if it is a lower score.

RECITATION GRADING:

All recitation work should be able to be done during the recitation time. To allow extra time for students to complete the work, the recitation assignment may be turned in up to the Sunday at 5pm following the recitation. As assignments can be completed in class, **No late assignments will be accepted.**

ASSIGNMENT GRADING LOGISTICS:

There are ten programming assignments in this course that are due on Fridays at 12:30 pm (just before class). You may automatically receive a 3-day extension for turning in the assignment with a 20% penalty. No assignments are accepted after this 3-day period.

There are three assignments that require that you participate in a grading interview. In the interview, you will be asked questions about the assignment you submitted. These questions are designed to test your understanding of the code as well as provide you with an opportunity to show your TA what you know. Your TA will announce when grading meetings are available, and it is your responsibility to schedule an interview with your TA as soon as the scheduler is posted on Moodle.

INTERVIEW GRADING GUIDELINES:

This is a large class, and as such, it is important that every student attends their grading meetings at the scheduled time. Please take the time to read and understand the following interview grading guidelines.

- Grading meetings are 10-15 minute appointments, scheduled Monday-Friday
- Sign-up is on Moodle
- If you need to reschedule your interview, that is okay, but you must email your TA at least 24 hours in advance (i.e. no waking up 5 minutes before the appointment, realizing you are going to be late, and sending a panicked email cancelling at the last minute). Emergency situations are an exception and will be evaluated on a case-by-case basis.
- There is a 1-minute "grace period" for being late, after that it is 10% off for each minute the student is late, at 6 minutes late you get a zero for the interview.
- My advice to all students is to get to the appointment 5 minutes early and use the extra time to prepare.

The assignments should always be submitted both on Moodle and to the auto-grader. If you forget to submit your assignment to Moodle you will receive a 10 point penalty. If you don't submit your assignment to both Moodle and the auto-grader you will get a zero.

The grades for this class follow the standard percentage breakdown for the College of Engineering:

93%-100%	A	73%-77%	C
90%-93%	A-	70%-73%	C-
87%-90%	B+	67%-70%	D+
83%-87%	B	63%-67%	D
80%-83%	B-	60%-63%	D-
77%-80%	C+	0%-60%	F

COURSE CALENDAR

WEEK 1, JANUARY 15-20

Topics: Introduction, Representation, Algorithms and Pseudo-code

WEEK 2, JANUARY 22-27

Topics: Problem decomposition, C++ programs, Variables, Data Types and console input/output

Due:

Recitation 1: VM setup, Accessing Moodle

Assignment 1: Algorithms and Pseudo-code

WEEK 3, JANUARY 29- FEBRUARY 3

Topics: Functions, calling functions, and branching

Due:

Recitation 2: Code Blocks, Submitting to Auto-grader

Assignment 2: Functions to implement algorithms

WEEK 4, FEBRUARY 5-10

Topics: Control structures – branches, loops and iteration

Due:

Recitation 3: Branching

Assignment 3: Functions for data calculations and string concatenation (*interview grading*)

WEEK 5, FEBRUARY 12-17

Topics: Loops and strings

Due:

Recitation 4: Strings

Recitation 5: Practice Midterm 1

Practicum: (during recitations)

Functions (declaration, parameters, return values), IF/IF-ELSE, Strings

WEEK 6, FEBRUARY 19-24

Topics: Arrays

Due:

Assignment 4: Functions using loops and branches

WEEK 7, FEBRUARY 26- MARCH 3

Topics: File I/O

Due:

Recitation 6: Iteration FOR/WHILE

Assignment 5: Functions to fill array, process data in an array

WEEK 8, MARCH 5-10

Topics: Classes

Due:

Recitation 7: Arrays

Assignment 6: Functions to read data from a file into an array, processing the data in an array,
writing the results to a file. *(interview grading)***WEEK 9, MARCH 12-17**

Topics: Class inheritance, method overloading, and encapsulation

Due:

Recitation 8: File IO

Recitation 9: Practice Midterm 2**Practicum: (during recitations)**

Iteration (WHILE/FOR), Arrays, File IO, Class declaration

WEEK 10, MARCH 19-24

Topics: Putting it all together and Advanced C++ topics

Due:Assignment 7: Class definitions, constructors, public/private data and methods, methods for data
access and analysis.**WEEK 11, MARCH 26-31**

No class; Spring Break

WEEK 12, APRIL 2-7

Topics: C++ advanced topics and Project

Due:

Recitation 10: Class abstractions

Assignment 8: Proposal for Final Project in C++

WEEK 13, APRIL 9-14

Topics: Python variables, functions, control structures, lists

Due:

Recitation 11: Classes with arrays, Arrays of classes

Recitation 12: Practice Midterm 3

Practicum: (during recitations)

File IO, Classes, and Arrays of user-defined objects

WEEK 14, APRIL 16-21

Topics: Python Lists, Iteration, and Classes

Due:

Project: Final Project in C++ (*interview grading*)

Assignment 9: Python solution of C++ assignments

WEEK 15, APRIL 23-28

Topics: Python File IO, Dictionaries

Due:

Recitation 13: Python functions and lists

WEEK 16, APRIL 30 - MAY 5

Topics: Python Exception handling, Advanced Topics

Due:

Recitation 14: Python dictionaries and classes

Recitation 15: Practice Midterm 4

Assignment 10: Python Data analysis project

Practicum: (during recitations)

Python control structures, functions, lists, and dictionaries

FINAL EXAM WEEK

Due:

Exam Date and Time:

Tuesday, May 9, 1:30pm

This final exam time can be used to replace your previous midterm scores. You will be able to take a replacement practicum for any of the four midterms. If you attempt any of the exams, your new score for that practicum will *replace* your previous score (*even if it is lower*). If you are satisfied with your midterm grades, you do not need to take any final exam.

SUBMITTING WORK LATE

You can receive a three-day extension on any assignment or recitation for a 20% grade penalty. After three days, your homework is late and cannot be turned in. In the event of a documented personal, family, or medical emergency, consult your TA about receiving a penalty free extension. If you know you will be missing a weekly recitation, go to a recitation with the same TA being held at a different time.

OTHER ASSIGNMENT INFORMATION

Written work must be neat and readable, with adequate spacing and margins. Your name, the date, and your section number must be at the top right of the first page. Code files should have your name, date, and homework number included as comments at the top of the file.

A limited amount of printing may be required in this class. You need to ensure that your printing account has sufficient funds for this. Your initial allocation may deplete quickly, depending on your other printing activities. If this causes problems, please come see me.

ATTENDANCE

Attendance at all class meetings and recitations is highly recommended. You are responsible for knowing the material presented during class and recitation, even if you were not in attendance when the material was presented.

CLASSROOM BEHAVIOR

Students and faculty each have responsibility for maintaining an appropriate learning environment. Those who fail to adhere to such behavioral standards may be subject to discipline. Professional courtesy and sensitivity are especially important with respect to individuals and topics dealing with differences of race, color, culture, religion, creed, politics, veteran's status, sexual orientation, gender, gender identity and gender expression, age, ability, and nationality. Class rosters are provided to the instructor with the student's legal name. I will gladly honor your request to address you by an alternate name or gender pronoun. Please advise me of this preference early in the semester so that I may make appropriate changes to my records. For more information, see the policies on [class behavior](#) and [the student code](#).

It is my expectation that each of you will be respectful to your fellow classmates and instructors at all times. In order to create a professional atmosphere within the classroom, you are expected to:

- * Arrive to class on time.
- * Turn off your cell phone (talk and text).
- * Remain for the whole class; if you must leave early, sit in the back and leave without disrupting others.

- * Bring your laptop to class if you have one to participate in classroom activities. Please restrict laptop use to these activities only, no email, Facebook, Youtube, etc.
- * Put away newspapers and magazines.
- * Refrain from having disruptive conversations during class.
- * Display professional courtesy and respect in all interactions related to this class.

Compliance with these expectations will assist all of us in creating a learning community and a high quality educational experience.

Though many of the above stated policies address academic climate within the classroom, these policies should also be upheld outside of the classroom. As a member of the CU community you are expected to consistently demonstrate integrity and honor through your everyday actions. Faculty, TAs, and staff members are very willing to assist with your academic and personal needs. However, multiple professional obligations make it necessary for us to schedule our availability. Suggestions specific to interactions with faculty and staff include:

- * Respect posted office hours. Plan your weekly schedule to align with scheduled office hours.
- * Avoid disrupting ongoing meetings within faculty and staff offices. Please wait until the meeting concludes before seeking assistance.

EMAIL POLICY

Respect faculty and staff policies regarding use of email and note that staff and faculty are not expected to respond to email outside of business hours. Send email messages to faculty and staff using a professional format.

Emails in reference to this course must follow the following tips for a professional email or they will be ignored:

- * Always fill in the subject line with a topic that indicates the reason for your email to your reader. (Use CS1300 in your emails for this course)
- * Respectfully address the individual to whom you are sending the email (e.g., Dear Professor Smith).
- * Avoid email or text message abbreviations.
- * Be brief and polite.
- * Add a signature block with appropriate contact information.
- * Reply to email messages with the previously sent message. This will allow your reader to quickly recall the questions and previous conversation.

CSCI 1300 SPRING 2016 COLLABORATION POLICY

The Computer Science Department at the University of Colorado at Boulder encourages collaboration among students. To support students in collaboration the Department has created a Collaboration Policy that makes explicit when their collaborative behavior is within the bounds of collaboration and when it is actually academic dishonesty, and therefore a violation of the University of Colorado at Boulder's Honor Code.

Students are most successful when they are working with other students to understand new concepts. The ultimate goal is that you fully understand the code you develop and be able to collaborate with others in a mutually beneficial way.

Unless otherwise specified, you may make use of outside resources (internet, other books, people), but then you must give credit by citing your sources in the comments inside your code. Use of outside resources does not include downloading complete, or almost complete, solutions to an assignment, whether you cite the source of the solution or not. This is considered plagiarism and violates the University's Honor Code policy.

Examples of citing sources include:

```
// Modified version from https://github.com/Phhere/MOSS-PHP
// Adapted from Program #7.2 in book "Accelerated C++" by Stroustrup
// Worked with Joe Smith from class to come up with algorithm for sorting
// Received suggestions from stackExchange website (see http://....)
```

A good rule of thumb: "If it did not come from your brain, then you need to attribute where you got it."

Collaboration Exceptions

Certain homework, quizzes, or exams may be required to be completed without outside resources (see course overview for details). In these cases it is your responsibility to know the extent of approved resources and use only those that have been specifically allowed. Use of outside resources in these cases would violate the collaboration policy.

Examples of violating the Collaboration Policy (resulting in a 0 score for assignment)

- Sharing a file with someone else.
- Submitting a file that someone else shared with you.
- Stealing a copy of someone else's work and submitting as your own (even with modification).
- Copying or using outside resources to solve a component of a larger problem and not citing your sources.
- Copying or using an entire solution that you didn't generate, regardless of whether you cited your sources.

Examples of collaborating correctly:

- Asking another student for a helpful suggestion.
- Reviewing another student's code for issues/bugs/errors.
- Working together on the whiteboard (or paper) to figure out how to approach and solve the problem. In this case you must include that person's name in your collaboration list at the top of your submission.

One way to know you are collaborating well is if everyone fully understands the code that is developed. If you do not understand what is in your code or why certain parts of the code are included, you need to ask someone to clarify! This collaboration policy requires that you be able to create the code (or solve the problem) on your own before you submit your assignment.

Any discovered incidents of violation of this collaboration policy will be treated as violations of the University's Academic Integrity Policy and will lead to an automatic academic sanction in the course and a report to both the College of Engineering and Applied Science and the Honor Code Council. Students who are found to be in violation of the Academic Integrity Policy can be subject to non-academic sanctions as well, including but not limited to university probation, suspension, or expulsion.

Other information on the Honor Code can be found at www.colorado.edu/policies/honor.html and www.colorado.edu/academics/honorcode.

Collaboration boundaries are hard to define crisply, and may differ from class to class. If you are in any doubt about where they lie for a particular course, it is your responsibility to ask the course instructor.

DISCRIMINATION AND HARASSMENT

The University of Colorado Boulder (CU-Boulder) is committed to maintaining a positive learning, working, and living environment. CU-Boulder will not tolerate acts of discrimination or harassment based upon Protected Classes or related retaliation against or by any employee or student. For purposes of this CU-Boulder policy, "Protected Classes" refers to race, color, national origin, sex, pregnancy, age, disability, creed, religion, sexual orientation, gender identity, gender expression, veteran status, political affiliation or political philosophy. Individuals who believe they have been discriminated against should contact the Office of Discrimination and Harassment (ODH) at 303-492-2127 or the Office of Student Conduct (OSC) at 303-492-5550. The [full policy on discrimination and harassment](#) has more information.

DISABILITY ACCOMMODATIONS

If you qualify for accommodations because of a disability, please submit to your professor a letter from Disability Services in a timely manner (for exam accommodations provide your letter at least one week prior to the exam) so that your needs can be addressed. Disability Services determines accommodations based on documented disabilities. Contact Disability Services at 303-492-8671 or by e-mail at dsinfo@colorado.edu. If you have a temporary medical condition or injury, see the Temporary Injuries guidelines under the Quick Links at the Disability Services website and discuss your needs with your professor.

RELIGIOUS OBSERVANCES

Campus policy regarding religious observances requires that faculty make every effort to deal reasonably and fairly with all students who, because of religious obligations, have conflicts with scheduled exams, assignments, or required attendance. In this class, (insert your procedures here). See the campus policy regarding religious observances for full details.

SEXUAL MISCONDUCT, DISCRIMINATION, HARASSMENT AND/OR RELATED RETALIATION

The University of Colorado Boulder (CU Boulder) is committed to maintaining a positive learning, working, and living environment. CU Boulder will not tolerate acts of sexual misconduct, discrimination, harassment or related retaliation against or by any employee or student. CU's Sexual Misconduct Policy prohibits sexual assault, sexual exploitation, sexual harassment, intimate partner abuse (dating or domestic violence), stalking or related retaliation. CU Boulder's Discrimination and Harassment Policy prohibits discrimination, harassment or related retaliation based on race, color, national origin, sex, pregnancy, age, disability, creed, religion, sexual orientation, gender identity, gender expression, veteran status, political affiliation or political philosophy. Individuals who believe they have been subject to misconduct under either policy should contact the Office of Institutional Equity and Compliance (OIEC) at 303-492-2127. Information about the OIEC, the above referenced policies, and the campus resources available to assist individuals regarding sexual misconduct, discrimination, harassment or related retaliation can be found at the OIEC website.

HONOR CODE

All students enrolled in a University of Colorado Boulder course are responsible for knowing and adhering to the academic integrity policy of the institution. Violations of the policy may include: plagiarism, cheating, fabrication, lying, bribery, threat, unauthorized access, clicker fraud, resubmission, and aiding academic dishonesty. All incidents of academic misconduct will be reported to the Honor Code Council (honor@colorado.edu; 303-735-2273). Students who are found responsible for violating the academic integrity policy will be subject to nonacademic sanctions from the Honor Code Council as well as academic sanctions from the faculty member. Additional information regarding the academic integrity policy can be found at honorcode.colorado.edu.