

CS1428

Foundations of Computer Science: I

Instructor: Gentry Atkinson (he/him)

Term: Summer 2023

Email: gma23@txstate.edu

Section 501:

Class Meeting Time: MTWRF 12:00-1:40 PM

Class Location: Derrick 240

Virtual Office Hours:

TBD

Important Dates:

May 29 th	Memorial Day
May 30 th	First day of Class
May 31 st	Last day to add a class
June 2 nd	Last day to drop with no record
June 23 rd	Deadline to apply for graduation
June 30 th	Final Exam, 2:00-4:30 PM

Course Description: Introductory course for computer science majors, minors and others desiring technical introduction to computer science. Contains overview of history and structure of the digital computer, including binary data representation. Problem solving, algorithm development, structured programming, good coding style, and control structures of C++ are emphasized.

Prerequisites: MATH 1315 or MATH 1317 or MATH 1319 or MATH 1329 or MATH 2417 or MATH 2471, with a grade of "C" or better, or ACT Mathematics score of 24 or more, SAT Mathematics score of 520 or more, SAT Math Section score of 550 or more.

Course Format:

- This class will held in-person but some lectures may be posted as videos in the event of time conflicts or to provide supplemental content.
- Exams, quizzes, and assignments will be delivered through the Canvas LMS. Students will need access to reliable internet to complete homework assignments.

Learning Objectives:

Students who successfully complete this course will...

- Be able to use fundamental programming constructs: syntax and semantics of C++, variables, types, expressions, assignment, I/O, control structures, functions, and arrays.
- Know how to apply algorithms for problem solving.
- Be able to use fundamental data structures: primitives, arrays, and strings.
- Be familiar with machine-level representation of data, including binary numbers and non-numeric data.
- Apply user-defined data types including structures.
- Understand the history and structure of digital computers.
- Analyze and explain the behavior of simple programs.
- Describe and apply strategies for effectively debugging programs.
- Use pseudo-code to design problem solving strategies.
- Modify and expand short programs.
- Install and use an integrated development environment.

Class Schedule:

Lecture 1	Introduction
Lecture 2	What is Computing?
Lecture 3	Input/Output
Lecture 4	Branching
Lecture 5	Looping
Midterm Exam	
Lecture 6	Functions
Lecture 7	Arrays
Lecture 8	Passing by Reference
Lecture 9	Structs
Lecture 10	Final Review
Final Exam	

Grading Policy:

- In-class assignments: 10%
- Check on learning quizzes: 10%
- Midterm: 15%
- Final: 15%
- 4 coding projects: 40%
- Lab Grade: 10%

Extra Credit:

Completing one challenge on [HackerRank](#) and submitting a screenshot on Canvas will be worth one point on a student's final grade. This can be done up to three times.

The specific problems will be posted in the Assignments section of Canvas.

Attendance Policy:

Students are encouraged but not required to attend the daily lectures. Lecture slides will be posted on Canvas, but not videos of live lectures. Students who choose not to attend lectures are still expected to learn all of the material presented in this class and may have difficulty completing assignments.

Some lectures may be moved to an asynchronous video format in the event of emergency or to accommodate the instructor's travel schedule. No more than four lectures will be delivered by video this semester.

Late and Make-up Work:

Make-up exams will only be given for students who arrange an absence ahead of time or who are given an excused absence by the Dean of Students. Students who know that they are not going to be available during an exam time should contact the instructor to arrange an alternative testing time as soon as they find out about the conflict.

Students who experience illness or personal tragedy that prevents them from submitting work should contact the Dean of Students to request an excused absence. Students who are given an excused absence by the Dean do not need to explain the nature of the illness or tragedy to the instructor and will be given an opportunity to submit any work or exam that was due during the excused period.

The four Coding Projects each have a 24 hour period for late submissions. Projects submitted during the late submission period will be penalized by 20%.

Students are always welcome to request additional working time on an assignment or quiz (other than the midterm and final exams) through email, but these will be considered on a case-by-case basis.

Required Materials:

- The textbook for this class will be **The C++ Workshop** by Dale Green, Kurt Guntheroth, and Shaun Ross Mitchell which is available free through Alkek library. Each lecture will have an accompanying reading list.
- Students will need access to a computer with a web browser and an Integrated Development Environment (IDE). All class coding demonstrations will be done using the **CodeBlocks** IDE. Students may alternately install the **Xcode** (Macintosh) IDEs on their personal computers, or use the **Replit** browser based IDE.
- A laptop or tablet will be necessary for the midterm and final exams.
- Students are encouraged but not required to bring a laptop or tablet to class for participating in live coding.

Course Format:

- The lectures for this class will be in person, and will generally not be available for asynchronous review.
- Lecture slides, assignments, and quizzes will all be posted on Canvas. Students should make sure that they have access to Canvas before attending this class.
- The midterm and final exam will be given **in person**. Students will need to bring a laptop or tablet with them on the days of those two exams.
- Attendance is encouraged but not required at the daily lectures. The daily coding problem is required whether the students attends lecture or not.
- Students must be enrolled in a lab section along with this lecture.

Contact and Response Time:

Students can contact their instructor outside of class by attending office hours or through email. Emails which are delivered between Monday at 8am and Friday at 5pm will be answered within 24 hours. Emails or posts outside of that window will be answered within 48 hours.

Please do not contact me through the Canvas messaging system. Use my actual email address if you need to reach me outside of class hours.

Accommodations for students with disability:

Any student with needs requiring special accommodations should contact the Office of Disability Services(ODS) at the LBJ student center (5-5.1). Students who qualify for extra time for exams must take their test with ATSD and must schedule their test at the same time the test is given in class. Note: you must submit your request with ATSD at least 2 business days before the exam date!

Students are never required or expected to discuss their disability with an instructor.

Communication Policy:

Students that need to contact the instructor outside of class and office hours should do so through email using the address at the top of this syllabus. I do not monitor and will not respond to communications through other channels (e.g. Canvas, Slack, Discord, Facebook). I will try to respond to emails that I receive Monday-Friday within 24 hours or within 48 hours for emails that I receive on Saturday or Sunday. Students should be aware that my responses might be slower before major deadlines.

Technical Difficulties :

If a failure of Canvas or the school's network interferes with an exam, a poll will be posted on Canvas to determine a "mitigation policy" for the grades on that exam. Options will include but are not limited to dropping a portion of the exam, applying a uniform curve to all exams, or re-taking the exam.

Network or software failures will not be considered when grading the Weekly Assignments, Coding Projects, and Independent Quizzes. Students responsible for finding access to a stable internet connection and for knowing how to use the IDE they have chosen to install. Computers with a wired internet connection and a working IDE maintained by the CS department are available in Derrick 231 and MCS 590.

Academic Honesty:

Students should remember that cheating only devalues their own education. You are expected to adhere to the University's Academic Honor Code:

<https://www.txstate.edu/honorcodecouncil/Academic-Integrity.html>

- You may work together on your programming assignments, in groups of 4 or less. If you submit a program that is the result of group work, you must list the names of all contributors in the file header. Each student must submit their own program, even if it is the same as another student's. The penalty for not citing collaborators will be -30 points for that assignment.
- Do not include code obtained from the internet or any other source in your programming assignment without properly attributing the original author (except what is provided by the instructor during the current semester). Do not post your solution anywhere on the internet. The penalty for either of these violations will be a 0 for that assignment.
- Submitting work done by others as if it were your own is an act of dishonesty.
- The penalty for violating academic honesty on a test or exam is a reduction of points up to a grade of 0 on the test or exam.
- Students who engage in dishonest activities will have to explain their actions to the Honor Code Council.

Drop Policy:

Be familiar with the [academic calendar](#) and be aware of the drop/add dates.

You must follow the withdrawal and drop policy set up by the University:

<http://www.registrar.txstate.edu/registration/dropping-or-withdrawing.html>

This document may be updated later to correct typos, clarify statements, or to adjust the schedule.