CS 1428: Foundations of Computer Science I
Spring 2020

Section 253

Instructor: Dr. Jill Seaman
Comal 210D
js236@txstate.edu

Course Webpage: http://www.cs.txstate.edu/~js236/cs1428

Office Hours: Tues, Wed, Thurs: 1:30pm – 3:00pm
and by appt.

Meeting Time/Place: Tues/Thurs 3:30PM-4:50PM Centennial Hall 157

Text: Tony Gaddis, Starting out with C++: From Control Structures through Objects, 9th Edition.

Note: we will be using the online **REVEL** version of this textbook. The course invite link is provided in an announcement on Canvas. Click on the link, then create a Pearson username and password. Then register either by using an access code you bought at the bookstore, or by purchasing immediate access using a credit card or PayPal. The printed version of the text is not required for the class.

List of required readings:
Chapters 1-7, 11.1-11.8 Readings will be assigned through the REVEL system.

Required In-Class Response system: We will be using the **Squarecap** classroom response system in class. You will be able to submit answers to in-class questions using a smartphone, tablet, or laptop. To sign up, visit **www.squarecap.com** on your web browser (Google Chrome is the preferred browser). Use your txstate email address as your username. Squarecap requires a paid subscription.

Course Description: Introductory course for computer science majors, minors and others desiring a technical introduction to computer science. The course emphasizes problem solving, algorithm development, structured programming, good coding style, and programming in C++.

Prerequisites: MATH 1315 or equivalent

Course Objectives:
At the end of the course, the students should be able to:

1. Describe the properties of good algorithms.
2. Design and develop good algorithms using a top-down approach.
3. Use the C++ programming language to implement, test, and debug algorithms for solving simple problems.

4. Explain the concepts of data types, variables, and literals and use them in programs.
5. Write C++ code that solves computational problems.
6. Use an if or if-else construct to implement branching in an algorithm.
7. Use a for loop for definite iteration.
8. Use a while or do-while loop for indefinite iteration.
9. Use functions and parameters to simplify longer programs and reuse code from previous solutions.
10. Demonstrate the mechanics of parameter passing with emphasis on the difference between pass by value and pass by reference.
11. Manipulate data in arrays.
12. Create a new data type by using a structure.
13. Analyze and explain the behavior of simple programs involving the fundamental programming constructs covered in this class.
14. Modify and expand short programs that use the constructs covered in this class.
15. Describe strategies that are useful in debugging.
16. Use a Windows- or Mac-based editor and compiler environment to develop programs in C++.

Grading:	Participation:	5%	Attendance + Squarecap
	REVEL	5%	From reading assignments
	Programming Assignments:	10%	lowest of 7 is dropped
	Lab:	10%	From your lab instructor
	Test 1:	15%	Feb 13 (Thur)
	Test 2:	15%	Mar 12 (Thur)
	Test 3:	15%	Apr 16 (Thur)
	Final Exam (comprehensive):	25%	May 12 (Tues) 2:00 - 4:30pm

Participation: Bring a web-enabled device to every class to access the Squarecap system. You will be asked questions in each class. (2 points each). We will also track attendance (2 points per day, 1 if late). You will need to scan your ID card in the classroom each day of class. The participation grade is computed as follows:
avg = (your squarecap points + your attendance points) / total possible points
participation = the minimum of: **avg***100 / 80 and 100%.
This means you can miss up to 20% of the points and still get 100%.
This 20% is a buffer that includes excused absences.

REVEL: There will be assigned reading from the book each week. These will be assigned in the REVEL system. You must complete the assigned reading and embedded quizzes by the deadline indicated.

Makeup Policy: Missed Revel assignments, Squarecap questions and programming assignments cannot be made up.

Final Exam Substitution: I will replace your minimum test score with the final exam score under the following circumstances:
1. You have an instructor-approved excused absence for that test.
2. You have not missed a test and your final exam score is greater than your minimum test score.

Notifications from the instructor: Notifications related to this class will be sent to your Texas State e-mail account. Be sure to check it regularly.

Canvas: We will use the following tools in Canvas this semester (NOT TRACS):

- Grades to track your grades
- Attendance to track your attendance
- Assignments to submit your Programming Assignments, and receive feedback/grading
- Files to post sample code from lecture

Everything else will be on the class webpage

Campus Labs: Use **Derr 231** or **MCS 590** to work on your programming assignments. You may also use your own computer, but you need to install CodeBlocks or some other C++ IDE first (such as Xcode for the Mac, or CLion). The lab instructors and tutors may be able to help you with the installation.

HELP: Other than the instructor's office hours, there are other places to obtain assistance:

Derr 231: Computer Science Department Lab Tutors are available in to help with your programming assignments.

CLC (Collaborative Learning Center): Free walk–in tutoring provided by students in the H–LSAMP Scholars Program. Ingram Hall #3202.

SLAC (Student Learning Assistance Center): walk-in tutoring lab and **Supplemental Instruction**: a nontraditional form of tutoring that focuses on collaboration, group study, and interaction for assisting students undertaking "traditionally difficult" courses. A trained peer who has successfully negotiated the course (the SI Leader) will facilitate 3 one-hour study sessions per week for group study.

TA's office hours see class website for times and location.

Withdrawals/drops: You must follow the withdrawal and drop policy set up by the University and the College of Science. You are responsible for making sure that the drop process is complete.
http://www.registrar.txstate.edu/registration/dropping-or-withdrawing.html

Last day to drop with automatic "W": March 31, 2020.

Classroom Behavior: My main rule is to not disrupt or distract other students during class! Additionally please treat other students and the instructors with respect. See the website for the Texas State Teaching Theater policies!!!

Accommodations for students with disability:
Any student with needs requiring special accommodations should contact the office of disability services 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!

Academic Honesty: You are expected to adhere to the University's Academic Honor Code as described here: 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 students'. The penalty for not citing collaborators will be -30 points for that assignment.

- Do not share your code electronically with anyone except someone you are working closely with, and whose name you will include as a co-author in your program header.

- Do not include code obtained from the internet or any other source in your programming assignment (except what is provided by the instructor during the current semester). The penalty for submitting a program that includes code from the internet or any other source outside of allowed resources 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.

Course Content: There are 7 main topics or units:

Unit 1: Intro to Programming and C++
Unit 2: Expressions and I/O
Unit 3: If/Else Switch
Unit 4: Loops
Unit 5: Arrays
Unit 6: Functions
Unit 7: Structures

For each unit I will provide the following:

- Reading assignments from the book (see REVEL).
- Squarecap questions
- Lecture slides
- A Programming Assignment

Exam coverage:

Test 1 covers Units 1-2.
Test 2 covers Units 3-4.
Test 3 covers Units 5-6.
Final Exam covers Units 1-7

Each Exam will have multiple choice questions (50%) and coding questions (50%)