CS 1428 - Foundations of Computer Science I  
Fall 2020

Catalog 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.

Prerequisite: 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 Objectives:
- Understand the history and structure of the digital computer.
- Explain the organization of the classical von Neumann machine and its major functional units.
- Understand binary data representation in the modern computer, including the representation of non-numeric data.
- Understand that fixed-length number representations affect accuracy and precision.
- Identify the necessary properties of good algorithms.
- Discuss the importance of algorithms in the problem-solving process.
- Understand the software development process, good coding style, and algorithm development.
- Use pseudo-code or a programming language to implement, test, and debug algorithms for solving simple problems.
- Introduce the syntax of the C++ programming language.
- Understand how to use an if or if-else construct to implement a branch in an algorithm.
- Understand how to use a for loop for definite iteration.
- Understand how to use a while or do-while loop for indefinite iteration.
- Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
- Describe the mechanics of parameter passing with emphasis on the difference between pass by value and pass by reference.
- Manipulate data in arrays.
- Create a new data type by using a structure.
- Analyze and explain the behavior of simple programs involving the fundamental programming constructs covered by this unit.
- Modify and expand short programs that use standard conditional and iterative control structures and functions.
- Describe strategies that are useful in debugging.
- Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.
- Students will be able to use a Windows-based editor and compiler environment.

Course Notes:
COSC 2320
CS 1428 is the required introductory course for Computer Science majors and minors and will include a lab. (This course is intended for CS majors and minors only; non-CS majors should check their degree requirements or get with the academic advisor for an appropriate computer science class).

“Do you know the difference between education and experience?

Education is when you read the fine print.  
Experience is what you get when you don’t.”

— Pete Seeger
Visit the Texas State home page to get the latest updates on campus operations and COVID-19 related resources.

Food for thought:
Online courses are not for the 'faint of heart'. Successfully completing the requirements of a course where content is delivered remotely places more responsibility on the shoulders of the student. Learning is always a ‘participation sport,’ and staying engaged is vital to the learning process. Practice is especially important when learning to write efficient, readable code that is easily maintainable.

For most of you, C.S.1428 is a prerequisite for another course in your degree plan; so, it is especially important for you to digest the concepts presented in this course to establish a strong foundation before moving on.

If

- time management is one of your weaknesses,
- you are prone to procrastinate,
- you are easily distracted,
- you have difficulty completing tasks,
- your internet connection is weak or unstable, or
- you do not have access to the required hardware/software to complete the course work,

then an online environment might not be your best option.

For this class, you are required to have a working camera and a working microphone especially on exam or quiz days. You will be required to leave your camera on during Zoom lecture meetings. To receive any credit for proctored exams/quizzes, you will be required to leave both camera and microphone on for the duration of the exam/quiz.

For lecture classes, we will be using a combination of Zoom, TRACS and my faculty web site for course content delivery, most assessments, and/or assignment submission.

- I teach three sections of C.S.1428.
- You may only attend Zoom meetings for the section in which you are registered.
- Lectures will not be recorded.
- Office hours will be held via Zoom as well.
- You are required to install Code::Blocks on your personal device, the IDE used in the course for both lab and lecture assignments. Make sure to install the same version as the one installed in our labs. (If you have a Mac, you can install Xcode instead.)
- You are responsible for making any adjustments to accommodate differences between IDE requirements before submitting programming assignments, since programming assignments will be graded using the version, with the same settings, of Code::Blocks installed in our labs.
- You are responsible for proper installation of the IDE and for working out any bugs on your own system.

C.S.1428 labs are a required part of the course.

- Except for L29, L30, and L31, all other lab sections will be online only. L29, L30, and L31 meet on Fridays in Derrick 236 where an A/B "face to face“ model will be implemented. (Refer to your class schedule.) Your lab instructor will have more detail.
- Labs meet for the first time the week of August 31. (Installation of Code::Blocks is discussed as part of that first meeting.)

Your lab instructors will provide you with their contact information and their lab expectations.

Think ahead. Plan for your online class, especially on exam/quiz days. You should have a quiet place where you can listen to the presentation of new material, participate in class discussions or take an online proctored exam/quiz, a place free of distraction and free of any other people and/or pets whenever possible. Let your parents, roommates or friends know what time you need their cooperation. Consider moving your pet(s) into another room during meetings. Consider putting a ‘Do Not Disturb’ sign on the door. Turn off the ringer on your phone.
Zoom meetings minimum requirements:
- You are expected to behave in a professional manner when participating in Zoom meetings.
- Verbal exchanges should be respectful and profanity free.
- Backgrounds (visible through your camera) should be appropriate.
- Dress appropriately!
- For identification purposes, you are required to display your first and last names as provided to the university along with your profile picture for use by the host and/or co-host(s). Your picture should be a headshot of you and only you, one where you are easily identified. (no costumes, masks or images of superheroes and the like)
- Remember that you are expected to leave your camera on during the meeting.

You will receive e-mail notification that will include URLs needed to join Zoom meetings for
- C.S.1428 lectures.
- my Monday/Wednesday office hours.
- my Friday office hours.
- the final exam.
(Keep this e-mail in a safe place or two for easy access throughout the semester.)

TIME:
Section .001 09:00 a.m. - 09:50 a.m. MWF (Zoom meeting)
Section .002 10:00 a.m. - 10:50 a.m. MWF (Zoom meeting)
Section .007 11:00 a.m. - 11:50 a.m. MWF (Zoom meeting)

INSTRUCTOR: Becky Reichenau
OFFICE: Comal 210A

CS DEPT: Comal 211
E-MAIL: br02@txstate.edu
(You can expect a reply to your e-mail within 48 hours if you have used your Texas State e-mail address; however, do not depend on responses at night, on holidays or over a weekend.)

WEB PAGE: https://userweb.cs.txstate.edu/~br02/
(TRACS will be used to submit programming assignments. Some assessments (especially major exams) will be administered using TRACS. Some announcements may be posted to TRACS; however, most communication outside class meetings will be via Texas State e-mail accounts.)

OFFICE HRS: (tentative)
M/W 12:30 p.m.-02:00 p.m. via Zoom
F 12:30 p.m.-01:30 p.m. via Zoom

Others by appointment.
(Appointments need not be made during regularly scheduled office hours.)

(A waiting room will be enabled. Students will be called in on a first-come-first-served basis. If others are waiting, you will be limited to 10 minutes, so come prepared. If you still have questions, you are always welcome to move back to the waiting room to ‘get back in line’.)

NOTE: Your lab instructors and my student assistants will hold office hours via Zoom as well. Expect URLs from each of them individually.

TEXTBOOK: Gaddis, Tony  Starting Out with C++: From Control Structures through Objects, 9th Edition

SUPPORT MATERIALS:
- Instructor’s Web Site: https://userweb.cs.txstate.edu/~br02/
  (Access to detailed support files and assignments will be provided in the initial class meeting.)
GRADING POLICY: Quizzes/Daily Assignments 10%
Lab 15%
Programs 20%
2 Major Exams:
  Exam I 10%
  Exam II 20%
Final Exam (comprehensive) 25%

Note: You are required to show your Texas State student (photo) ID to your instructor/instructor’s assistant on exam days, so have it readily available. A driver’s license is not adequate. Exam scores will be recorded as zeros until your Texas State student ID is presented. Exceptions for students new to Texas State must be arranged at least 24 hours prior to an assessment.

**Quizzes over recently covered material are not typically announced in advance. Expect one every day, and you will always be prepared. Attendance quizzes will typically be given at the end of lecture. Content quizzes will typically be given at the beginning of lecture. Students who join the Zoom meeting late will not be given additional time to complete a quiz that was administered that day. Quizzes over lecture material covered during a Zoom meeting cannot be made up.**

Major exams will be announced at least one week in advance. They are typically scheduled during the equivalent of the sixth and eleventh weeks of a long semester; however, the actual dates may be adjusted to benefit the students.

Quizzes and/or major exams administered on TRACS will be proctored in the Zoom meeting scheduled on quiz/exam day. Therefore, to receive any credit, you are required to join the Zoom meeting and stay for the entire time while completing such assessments.

FINAL EXAM SCHEDULE:
C.S.1428.001 (MWF 9:00 a.m.) 08:00 a.m. - 10:30 a.m. Monday, December 7
C.S.1428.002 (MWF 10:00 a.m.) 11:00 a.m. - 01:30 p.m. Monday, December 7
C.S.1428.007 (MWF 11:00 a.m.) 08:00 a.m. - 10:30 a.m. Wednesday, December 9

Final exams will be administered only on the day and at the time indicated in the university exam schedule unless otherwise indicated above.

GRADING SCALE:

Determination of letter grade in the course:

<table>
<thead>
<tr>
<th>Semester Average</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;= 89.5</td>
<td>A</td>
</tr>
<tr>
<td>79.5 &lt;=</td>
<td>B</td>
</tr>
<tr>
<td>69.5 &lt;=</td>
<td>C</td>
</tr>
<tr>
<td>59.5 &lt;=</td>
<td>D</td>
</tr>
</tbody>
</table>

***Programs are very important to this course. Programs that contain compilation errors receive no credit. Programs that produce incorrect output will automatically receive a 30% penalty. To receive any credit for a programming assignment, the source code (along with appropriate support files) must be submitted electronically.***

Alert: Time permitting programs will be run through an Internet service designed for detecting plagiarism in software code.

LATE POLICY: All assignments are due by a specific time on a specific date. No late assignments will be accepted unless official (an original, not a copy) documentation can be provided that may warrant an extension. Regardless of the deadline set, there will always be people who want it extended.

NO OTHER "LATE" ASSIGNMENTS WILL BE ACCEPTED!!! NO EXCEPTIONS!!

ALLOW FOR NATURAL DISASTERS! The computer system used may ‘go down’, you overslept, your bus was late, you misread the due date, flash drives may have been left ‘somewhere’, etc. These types of events do NOT serve as an excuse for late work.

Now that the deadline for credit has been established, please respect this deadline, and plan accordingly.
If you take paperwork meant for your lecture instructor to the Computer Science Department office (Comal #211), you should send an e-mail to your instructor indicating that you have done so.

**GRADE DISPUTES:**
For complete discussion and possible resolution, grade disputes must be handled in a Zoom meeting during regularly scheduled office hours or by appointment when all records are readily available to your instructor. Disputes over grades must be discussed within **one week** from when grades are posted. For example, you must discuss programming assignment grades within one week of when feedback is posted on TRACS. This means within **one week** from the date the assignment feedback is posted. It does NOT mean one week from the date you decide to review the feedback.

**ACADEMIC OFFENSES:**
All assignments submitted for a grade should reflect the work of the individual student unless otherwise established in writing by your instructor. Violations will be dealt with according to Academic Procedures and Policies as outlined in the Texas State Student Handbook. Go to [http://www.dos.txstate.edu/handbook.html](http://www.dos.txstate.edu/handbook.html), and click on Academic Honor Code to review Academic Offenses and the Penalties for Academic Dishonesty.

**ATTENDANCE POLICY:** Attending lectures via Zoom meetings is **highly recommended**. You will be held responsible for material covered in those meetings should you choose not to attend. Some of the material covered in the meetings may not be readily available elsewhere. You are responsible for obtaining assignments and notes from fellow classmates for meetings you are unable to attend. I recommend that you obtain contact information from several classmates for those ‘rainy days.’

E-MAIL notifications related to this class will be regularly sent to your Texas State e-mail account. If you do not check it on a regular basis, forward your Texas State e-mail to an e-mail account that you do check on a regular basis.

**ABSENCE POLICY:** There will be **NO** make-up quizzes or exams. If you are absent at the time of a quiz or exam, a grade of zero will be recorded. Even so, it is a good idea to contact your instructor if such an absence occurs.

**DROP POLICY:** (Refer to the Academic Calendar).
- Automatic “W” deadline and last day to drop a class is Monday, October 26.
- Last opportunity to withdraw from the University is Thursday, November 19.
- Students who withdraw from the University after the automatic “W” date will be assigned a "W" or an "F"/"U" based on class performance up to that point in the semester. A "W" will be assigned only if the class average is passing on the day the withdrawal procedure is officially completed.

**Note:** Contact the Registrar’s Office as to the proper procedure to complete the drop/withdrawal process successfully. If you decide to withdraw from the University after the automatic “W” date previously mentioned, be sure to check with your instructor prior to completing the withdrawal procedure in order to verify whether you will be assigned a “W”, “F” or “U”. Contact the Registrar if you have further questions.

It is your responsibility to make sure the drop/withdrawal process is complete. Do not come to me later and say that you "thought” you had dropped but the process did not "go through" expecting me to change a grade of ‘F’ or ‘U’ to a ‘W’. Be sure to check your revised schedule to make sure the course dropped is no longer listed.

**ADA Compliance:** Students with special needs as documented by the Office of Disability Services who require accommodations should identify themselves to the instructor as soon as possible but **no later than the 12th class meeting in a long session** and **no later than the 4th class meeting during a regular summer session**. Students with special needs who have not already done so will be required to contact the Office of Disability Services to establish accommodations. Every effort will be made to secure the necessary accommodations to facilitate students with special needs/disabilities to enhance their performance in the classroom.

---

**Be a yardstick of quality. Some people aren’t used to an environment where excellence is expected.**

Steve Jobs
# TENTATIVE SEMESTER SCHEDULE

(Material covered and/or associated assignments may be adjusted throughout the semester to enhance student learning.)

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Topic(s)</th>
<th>Assignments</th>
</tr>
</thead>
</table>
| Week 1 | Aug 24    | - Course Introduction / Instructor’s Web Site  
- Computing Environment/Computer memory                                                      |             |
|       | Aug 26    | - Programming Process/C++ Program Structure/Documentation  
- Code::Blocks IDE;  
- Console I/O & Assignment statements / Syntax Errors/Debugging |             |
|       | Aug 28    | - Primitive data types / Variables / Named constants / Literals                     |             |
| Week 2 | Aug 31    | - Interactive program (e.g. age)                                             |             |
|       | Sept 2    | - Expression evaluation                                                      | Asg 0 Due   |
|       | Sept 4    | - Operators, precedence, expression types, type casting  
- Shorthand operators, pre-increment vs post-increment |             |
| Week 3 | Sept 7    | - Labor Day - Holiday                                                        |             |
|       | Sept 9    | - Formatting Output                                                         |             |
|       | Sept 11   | - File I/O                                                                 |             |
| Week 4 | Sept 14   | - Conditional Statements  
- Relational Operators  
- if (single-sided alternatives) |             |
|       | Sept 16   | - Conditional Statements  
- if else (double-sided alternatives) | Asg 1 Due   |
|       | Sept 18   | - Conditional Statements  
- if else-if else (multi-way branch) / Logical Operators / Short-circuit evaluation |             |
| Week 5 | Sept 21   | - Conditional Statements  
- Nested Logic |             |
|       | Sept 23   | - Conditional Statements  
- switch |             |
|       | Sept 25   | - Integer storage in memory, time permitting |             |
| Week 6 | Sept 28   | - Looping Control Constructs  
- for (count) loops |             |
|       | Sept 30   | - Looping, continued  
- nested for (count) loops | Asg 2 Due   |
|      | Oct 2     | - Exam I                                                                   |             |
| Week 7 | Oct 5     | - Looping, continued / Simple Input Validation  
- while loops |             |
|       | Oct 7     | - Looping, continued  
- sentinel value while loops / eof controlled while loops |             |
|       | Oct 9     | - Looping, continued  
- do while loops |             |
| Week 8 | Oct 12    | - Introduction to Functions and Modular Programming  
- Topics covered for lectures on functions not necessarily in this order: |             |
|       | Oct 14    | - Typed vs Void Functions | Asg 3 Due |
|       | Oct 16    | - Function Prototype / Function Call / Function Definition |             |
| Week 9 | Oct 19    | - Pass by Value vs Pass by Reference |             |
|       | Oct 21    | - Scoping Rules |             |
|       | Oct 23    | - Function Overloading, time permitting |             |
| Week 10 | Oct 26   | - Returning multiple values from a function |             |
|       | Oct 28    | - Introduction to 1-D Arrays / array processing with loops | Asg 4 Due   |
|       | Oct 30    | - 1-D Arrays / array processing with loops, continued |             |
| Week 11 | Nov 2     | - 1-D arrays, continued  
- initializing arrays / strings as arrays of char / restrictions on array use |             |
|       | Nov 4     | - Functions & 1-D Arrays |             |
| Nov 6 | - Exam II |             |             |
| Week 12 | Nov 9    | - Parallel Arrays / 2-Dimensional Arrays |             |
|       | Nov 11    | - Parallel Arrays / 2-D Arrays, continued | Asg 5 Due   |
|       | Nov 13    | - Functions & 2-D Arrays |             |
| Week 13 | Nov 16 | Review prog6 concepts – 1-D, 2-D, parallel arrays; pass by reference, by value |
| Week 13 | Nov 18 | 1-D/2-D/Parallel Arrays Quiz OR guest speaker |
| Week 13 | Nov 20 | **Work on prog6** |
| Week 14 | Nov 23 | Structures |
| Week 14 | Nov 25-27 | **Thanksgiving Holiday** |
| Week 15 | Nov 30 | Structs quiz (breakout rooms) |
| Week 15 | Dec 2  | **Last Day CS1428 Meets - Recap & Review** |
| Week 15 | Dec 4 | ‘Reading Day’; Final Exams Begin @ 5:00 p.m. |
| Week 16 | Dec 7–11 | **Final Exams** |