



CSCI 1210: Introduction to Computer Science II

2021 Winter Session	
Total Class Sessions: 25 Class Sessions Per Week: 6 Total Weeks: 4 Class Session Length (Minutes): 145 Credit Hours: 4	Instructor: Staff Classroom: TBA Office Hours: TBA Language: English

Course Description:

This course will enable students use computers to solve problems. The following topics will include in this course: Introduction to Computers and C++ Programming, C++ Basics, and Flow of Control, Functions, I/O Streams as an Introduction to Objects and Classes, Arrays, Strings and Vectors, Pointers and Dynamic Arrays, Introduction to Defining Classes, Introduction to Linked Lists. This course will also talk about dynamic memory allocation and memory management.

Learning Objectives:

Upon completion of this course, students will be able to:

- Gain experience solving problems with computers
- Understand arrays by designing and implementing programs that search and sort arrays
- Understand pointers and dynamic memory allocation by designing and implementing programs using pointers and dynamic memory allocation
- Understand differences between C and C++ in the areas of strings, pass by reference/passing pointers, and structs by designing and implementing programs that use C strings, C++ strings, C language structs and classes
- Master dynamic memory allocation and memory management

Course Materials:

Walter Savitch. Problem Solving with C++, 10th Edition, Prentice Hall, 2018.

Course Format and Requirements:

The course will take place in a computer lab and the course format including lecture, programming project, and in-class discussion.

The specific topics that will be covered in the classes are listed in the course syllabus. The class period will consist of an active learning environment. During a majority of the class time, students will be actively working on problems under the instructor's guides.

Attendance:

Attendance will not be taken but is strongly recommended. Each student will have three allowed



absences and no grade deduction will be made for the first three absences. More than three unexcused absences will result in an automatic reduction in your participation grade, for instance from A- to B+. Your active participation in the class is expected and encouraged.

Course Assignments:

Quizzes:

There will be 5 quizzes this semester, given during the discussion sections. Each quiz will be on the material covered that week. There will be NO make-ups for quizzes for any reason. All of the quizzes will be closed book.

Midterm Exam:

The in-class, close-book and non-cumulative midterm exam will be given through this course. The midterm exam will be based on the knowledge covered in class. No excuse will be accepted if students do not have legitimate excuses for absence. Physician Statement is required for missing the exam due.

Weekly Projects:

There will be four hands-on projects based on course need. It will count for 40% of your grade for the course. The projects will enrich students' knowledge on writing large programs. The score will be given based on the correctness of the program.

Final Exam:

The final will be in-class, cumulative and close-book. The final exams will be based on concepts covered in class. Note that the final will not be taken during the normal class times. Exact time and location for final will be announced later.

Course Assessment:

Quizzes	10%
Weekly Projects	40%
Midterm Exam	20%
Final Exam	30%
Total	100%

Grading Scale (percentage):

A+	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F
98-	93-	90-	88-	83-	80-	78-	73-	70-	68-	63-	60-	<60
100	97	92	89	87	82	79	77	72	69	67	62	

Academic Integrity:

Students are encouraged to study together, and to discuss lecture topics with one another, but all other work should be completed independently.

Students are expected to adhere to the standards of academic honesty and integrity that are



described in the Chengdu University of Technology's *Academic Conduct Code*. Any work suspected of violating the standards of the *Academic Conduct Code* will be reported to the Dean's Office. Penalties for violating the *Academic Conduct Code* may include dismissal from the program. All students have an individual responsibility to know and understand the provisions of the *Academic Conduct Code*.

Special Needs or Assistance:

Please contact the Administrative Office immediately if you have a learning disability, a medical issue, or any other type of problem that prevents professors from seeing you have learned the course material. Our goal is to help you learn, not to penalize you for issues which mask your learning.

Course Schedule:

Week	Topics	Activities
1	<ul style="list-style-type: none">• Go through syllabus and introduction to the course• Introduction to computers and C++ programming<ul style="list-style-type: none">➤ Computer systems➤ Programming and problem-solving➤ Introduction to C++➤ Testing and debugging• C++ Basics<ul style="list-style-type: none">➤ Variables and assignments➤ Input and output➤ Data types and expressions➤ Simple flow of control• More flow of control<ul style="list-style-type: none">➤ Using Boolean expressions➤ Multiway branches➤ More about C++ loop statements➤ Designing loops	<ul style="list-style-type: none">▪ Quiz 1▪ Quiz 2▪ Weekly project
2	<ul style="list-style-type: none">• Procedural abstraction and functions that return a value<ul style="list-style-type: none">➤ Top-down design➤ Predefined functions➤ Programmer-defined functions➤ Procedural abstraction➤ Scope and local variables➤ Overloading function names• Functions for all subtasks<ul style="list-style-type: none">➤ Void functions➤ Call by reference parameters➤ Using procedural abstraction➤ Testing and debugging functions	<ul style="list-style-type: none">▪ Quiz 3▪ Weekly project▪ Midterm exam



	➤ General debugging techniques	
3	<ul style="list-style-type: none">• I/O streams as an introduction to objects and classes<ul style="list-style-type: none">➤ Streams and basic file I/O➤ Tools for stream I/O➤ Character I/O• Arrays<ul style="list-style-type: none">➤ Introduction to arrays➤ Arrays in functions➤ Programming with arrays➤ Multidimensional arrays• Strings and vectors<ul style="list-style-type: none">➤ An array type for strings➤ The standard string class➤ Vectors• Dynamic memory allocation	<ul style="list-style-type: none">▪ Quiz 4▪ Quiz 5▪ Weekly project
4	<ul style="list-style-type: none">• Memory management• Pointers and dynamic arrays<ul style="list-style-type: none">➤ Pointers➤ Dynamic arrays• Introduction to Defining Classes<ul style="list-style-type: none">➤ Structures➤ classesIntroduction to Linked Lists	<ul style="list-style-type: none">▪ Quiz 6▪ Weekly project▪ Final Exam