Syllabus CSE 030 01, Fall 2021

Instructor:
Ammon Hepworth

Designation:
CSE 030: Data Structures

Catalog Description:
Through the course of this semester, we will explore C++, object-oriented programming, recursion, analysis of algorithms, and data structures such as: vectors, queues, stacks, lists, trees and graphs.

Textbooks and Other Required Materials:
Required Text: Data Structures and Algorithms in C++ by Michael T. Goodrich, et. al, 2nd edition. (ISBN-13: 978-0470383278)

Suggested references:
http://www.cplusplus.com/
http://www.cplusplus.com/doc/tutorial/

Course Objectives:
1. Correctly use recursion to solve a problem with a binary search tree or graph
2. Determine the Big-O speed of an algorithm.
3. Correctly implement the right data structure for a given problem
4. Correctly determine the relative runtimes of different algorithms
5. Analyze unstructured problems and design computer solutions
6. Apply or a create suitable algorithm to solve a particular problem.

Program Learning Outcomes:
An ability to:
1. Apply knowledge of computing and mathematics appropriate to the discipline
2. Analyze a problem and identify the computing requirements appropriate for its solution
3. Design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs, and use current techniques, skill, and tools necessary for computing practice.
4. Apply mathematical foundations, algorithmic principles, and computer science theory to the modeling and design of computer based systems in a way that demonstrates comprehension of the trade-offs involved in design choices.

Prerequisites:
CSE20/21 combination or equivalent knowledge

Course Policies:

Please attend all lectures and the lab section which you are assigned. Your participation grade will be a direct reflection of your lab attendance.

For lab assignments, you may work together with other students if you wish or when assignment asks for explicit collaboration. Giving each other help in finding bugs and in understanding the assignment is encouraged. It is permissible to allow other students to see small portions of your code on-screen during lab, but you may not allow them to copy directly. In general, the deadline for submission for a lab will be ONE WEEK after it is posted. You will have a chance to resubmit for one more day for full credit provided your original submission was before the lab deadline.

Remember to be responsible, respectful and kind.

Academic Dishonesty Statement:

a. Each student in this course is expected to abide by the University of California, Merced's Academic Honesty Policy. Any work submitted by a student in this course for academic credit will be the student's own work.

b. You are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give "consulting" help to or receive "consulting" help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e mail, an e mail attachment file, a diskette, or a hard copy. Should copying occur, both the student who copied work from another student and the student who gave material to be copied will both automatically receive a zero for the assignment. Penalty for violation of this Policy can also be extended to include failure of the course and University disciplinary action.

c. During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

Disability Statement:

Accommodations for Students with Disabilities: The University of California Merced is committed to ensuring equal academic opportunities and inclusion for students with disabilities based on the principles of independent living, accessible universal design and diversity. I am available to discuss appropriate academic accommodations that may be required for student with disabilities. Requests for academic accommodations are to be made during the first three weeks of the semester, except for unusual circumstances. Students are encouraged to register with Disability Services Center to verify their eligibility for appropriate accommodations.

Topics:

C++: We will begin with a crash course in C++ programming, including keyboard and terminal I/O, file I/O, loops, logic constructs (if/then/else), and steadily build up our C++ knowledge as the semester progresses.

Object-Oriented Programming (OOP): Beginning with the basic concepts of OOP, we will explore the building blocks of the OOP paradigm - classes, templates, modular code, etc.

Recursion / Algorithmic Complexity: Part of writing good programs is knowing how good they are and where their weaknesses lie. We will explore the mathematical tools available to calculate how algorithms scale with the complexity of their input.

Data Structures: This is what the majority of the class focuses on, using the skills we learn in the first half to develop useful data structures for organizing and representing data to solve real problems.

Class/laboratory Schedule:

Lecture: M/W 4:30pm-5:45pm, Classroom: COB2 110
Lab: See class schedule for the day/time of the section you are assigned

Midterm/Final Exam Schedule:

Midterm: Oct 20 (In class)
Final: Friday, Dec 17 at 3:00pm – 6:00pm

Assessment/Grading Policy:

20% midterm
30% final exam
30% labs
15% homework
5% lab attendance

Grades: if you get 90% you'll get at least an A-, 80% will give you at least a B-, and 70% at least a C-.

Contact Information:

My email: **ahepworth@ucmerced.edu**
Office: SE2 278

I will try to answer your emails within 48 hours. However, I may not be able to answer emails after 5pm or on weekends/holidays.

TAs: Xianzhong Ding, **xding5@ucmerced.edu**
        Younce Yang, **syang126@ucmerced.edu**
        Nasit Sony, **nsony@ucmerced.edu**

Office Hours:

Instructor: Wednesday at 12:30-1:30pm, or by appointment **(see TA's for help with the labs)**
TA: Lab sessions will be used as office hours, or by appointment

Course Calendar:

Tentative schedule, subject to change

Week	Lecture - Monday	Lecture - Wednesday	Lab
01 (8/23 – 8/27)		No class	
02 (8/30 – 9/3)	1: Introduction to C++	2: More about C++	Lab 1: Intro to C++
03 (9/6 – 9/10)	**No class - Labor day**	3: Computer Basics & Analysis Tools	**No lab – Labor day**
04 (9/13 – 9/17)	4: Analysis Tools & Searching and Sorting	5: Sorting & Functions	Lab 2: Arrays
05 (9/20 – 9/24)	5.5: Sorting & Functions	6: Recursive Functions File I/O	Lab 3: Searching and Sorting Arrays
06 (9/27 – 10/1)	7: Pointers Structures	8: Dynamic Memory & Classes	Lab 4: Functions
07 (10/4 – 10/8)	9: Classes	10: Templates & Inheritance	Lab 5: Binary Search
08 (10/11 – 10/15)	11: Linked lists	12: Linked lists	Lab 6/7: Structures/Classes
09 (10/18 – 10/22)	13: Mid-term Review	**Mid-term**	Lab 8: Linked list
10 (10/25 – 10/29)	14: Queues	15: Trees	Lab 9: Stacks
11 (11/1 – 11/5)	16: Binary Trees	17: Dictionaries: Binary Search Trees	Lab 10: Queues
12 (11/8 – 11/12)	18: Maps: Hash Tables	19: Graphs	**No Lab - Veteran's Day**
13 (11/15 – 11/19)	20: Graphs	21: Graphs	Lab 11: Binary Trees
14 (11/22 – 11/26)	22: Vectors & Exceptions	**No class - Thanksgiving**	**No lab - Thanksgiving**
15 (11/29 – 12/3)	23: STL Map & STL Unordered Maps: Maps Using Hash Tables	24: Final Review	Lab 12: Hash Tables
16 (12/6 – 12/10)	25: Job talk	**No class (Final Prep)**	Lab 13: Templates
17- 12/17	**Final (Friday, Dec 17)**		