Objectives & Outcomes:

The objective in this class is for students to become familiar with Data Structures and algorithm design and analysis concepts. The student who completes this course:

- Will understand the fundamental data structures used in computer programs and their applications.
- Will be familiar with fundamental algorithm design techniques and the interaction between data structures and algorithms.
- Will develop skills to analyze and evaluate the performance of algorithm implementations; be familiar with basic mathematical tools.
- Will understand the issues involved in implementing algorithms and data structures in code.

Catalog Description

CPSC 2120 Algorithms and Data Structures 4(3) Study of data structures and algorithms fundamental to computer science; abstract data-type concepts; measures of program running time and time complexity; algorithm analysis and design techniques. Preq: CPSC 1020 with a C or better or CPSC 1070 with a C or better. Coreq: CPSC 2121.

Recommended Textbooks: We will not use a specific textbook in class. Some Useful Books:


Useful Readings: C & pointers, and Simple Makefiles, C++ Tutorial
Schedule: Click here to view a tentative course schedule; will include additional reading materials. (This will be updated continually; check often)

Tentative Outline of Topics:

- **Algorithm Design Techniques.** Divide and conquer, dynamic programming, greedy algorithms.
- **Analysis Techniques.** Induction, recurrences, amortization, modular arithmetic.
- **Sorting and Selection.** Insertion sort, bubble sort, merge sort, quicksort, heap sort, counting sort, radix sort,
- **Fundamental Data Structures.** Arrays, linked lists, stacks, and queues.
- **Hashing.** Strategies for collision resolution, applications of hashing (e.g., in security, data mining, and information retrieval).
- **Binary Search Trees and Related Structures.** BST balancing mechanisms, B-trees,
- **Priority Queues.** Binary heaps, applications in more advanced algorithms (e.g., shortest paths).
- **String Matching.** Hashing approaches, suffix arrays and suffix trees.
- **Graph Algorithms.** Depth-first and breadth-first search, shortest paths (Dijkstra’s algorithm), Spanning trees.

Tentative Grading (subject to change):

The final grade for the course will be computed based upon the following distribution (tentative):

- Labs — 20%
- Home works — 20%
- 2 Tests — 30%
- Comprehensive Final Exam — 30%

Final letter grades are not assigned according to an absolute scale. Based on final numeric scores, the instructor will draw cutoffs between letter grades as deemed appropriate.

Grade Appeal: Any grade challenges regarding exams, quizzes, exercises, or programs must be emailed to the instructor, with detailed justifications, within one week of the date the grades are posted.

Attendance

*Attendance in Lecture is expected, but not required; Lab attendance each week is mandatory.* You are responsible for any material covered in class. You are responsible for any announcements, assignments or assignment modifications that are announced in class whether you are present or not. This also applies to arriving late to class or leaving early. **We will have unannounced quizzes during class; 5 to 10 quizzes are expected during the semester. No make-up quizz will be given.**
Deadlines

Work is due at the specified deadline. **Late work will not be accepted.** If a project (homework, programming assignment, etc.) is only partially completed you should submit what you have by the deadline. You should expect very little to no credit for projects that still generate syntax errors.

**Instructor Late:** If the instructor is late for class, students are expected to wait for 15 minutes before they leave. Asynchronous Online Class

**Cancellation of Class:** If classes are cancelled by the university on the day of a scheduled test, we will reschedule the test. Asynchronous Online Class

Academic Integrity

As members of the Clemson University Community, we have inherited Thomas Green Clemson's vision of this institution as a "high seminary of learning". Fundamental to this vision is a mutual commitment to truthfulness, honor, and responsibility, without which we cannot earn the trust and respect of others. Furthermore, we recognize that academic dishonesty detracts from the value of a Clemson degree. Therefore, we shall not tolerate lying, cheating, or stealing in any form. Read the university academic integrity statement. Unless otherwise specified in an assignment, you are expected to work independently on projects and labs. **If you are in doubt about the type of help you may give or receive on a project: ASK!**

Additional Resources:

1. **ACM.** The Clemson student chapter of the Association for Computing Machinery (ACM) offers occasional tutorial sessions on basic topics like “how to use Linux” and “command-line tools”. Please feel encouraged to take advantage of these if needed.

2. **Tutoring.** Free, drop-in tutoring is available for this class through the Academic Success Center. For details on policies and drop-in session information, please see the ASC website at clemson.edu/asc.

3. **Clemson Spring Semester 2021 Academic Calendar**

   Aug. 15, M New Student Convocation
   Aug 15-16, M-Tu Late enrollment
   Aug 16, Tu University Convocation
   Jan 6, W Classes begin
   Jan 12, Tu Last day to register or add a class or declare Audit
   **Jan 20, W Last day to drop a class or withdraw from the University without a W grade**
   Jan 27, W Last day to apply for May Commencement
   **Feb 26, F Last day for instructors to issue midterm evaluations**
   Mar 12, F Last day to drop a class or withdraw from the University without final grades
Mar 15-19, M-F Spring break
Apr 12, M Registration for fall and summer terms begins
April 22-23, Th-F Classes meet; exams permitted in labs only
Apr 26-30, M-F Examinations
May 3, M 9:00 A.M.—Deadline to submit candidate grades
May 5, W 9:00 A.M.—Deadline to submit other grades
May 5, W Candidates for graduation may access grades
May 6, Th-F Commencement