CSc 140 - Advanced Algorithm Design and Analysis

3 Credits
Fall 2021

- *Administrative Information*

  1. **Instructor:** Parham Phoulady
  2. **Department:** Computer Science
  3. **Meeting Time and Location:** Monday/Wednesday, 4:00pm - 5:15pm – Online
     Synchronous using the pre-recorded videos
  4. **Office Hours:** Online on Zoom: Monday/Wednesday, 11:50am - 1:20pm; or by appointment
     Zoom meeting: `https://csus.zoom.us/j/96411569926`
  5. **E-Mail:** `phoulady@csus.edu`

- *Course Description*

  - Review of divide-and-conquer design and asymptotic notation. Algorithm design using dynamic programming, randomization and greedy methods. Analysis using recurrence relations and amortization. String and network-flow algorithms. NP-completeness, reductions, and approximation algorithms.

- *Prerequisites*

  CSc 130 - Data Structures and Algorithm Analysis.

  - *Basic Data Structures:* arrays, linked lists, queues, stacks, binary trees (including binary search trees)
  - *Some familiarity with Asymptotic Notation:* big-O notation
  - *Sorting:* insertion sort, bubble sort, quick sort, merge sort
  - *Proofs:* by contradiction, induction

- *Course Objectives*

  Students will be taught

  1. algorithm analysis techniques;
  2. how data structures relate to algorithm design and analysis;
  3. various algorithm design techniques, including, but not limited to: greedy, divide-and-conquer and dynamic programming;
  4. basic graph algorithms and their application to practical problems;
  5. the concept of NP-completeness and how to use reductions.

- *Textbooks*

  Required: "Introduction to Algorithms", 3rd Edition, by Thomas Cormen, Charles Leiserson, Ronald Rivest and Clifford Stein, MIT Press, 2009.

  Other suggested references:

  - "The Algorithm Design Manual", 2nd Edition, by Steven Skiena, Springer, 2012.
  - "Algorithm Design", by Jon Kleinberg and Éva Tardos, Pearson, 2006.
  - "Introduction to the Design and Analysis of Algorithms", 2nd Edition, by Anany Levitin, Pearson Adison Wesley, 2007.
  - "Introduction to Algorithms: A Creative Approach", 1st Edition, by Udi Manber.

- *Grading*

  - *Grading Rubric*

    1. Participation: 5%;
    2. Quizzes: 25%;
    3. Midterm: 35%;
    4. Final Exam: 35%;

  - *Grading Scale*
    Your weighted average for the course, using the above rubric, must be greater than or equal to <avg> in order to receive a grade of <grade>.

<avg>	<grade>
85.0%	A
70.0%	B
55.0%	C
40.0%	D
0%	F

  - The instructor **reserves the right** to make adjustments to the grading scale based on class averages.

  - The instructor **reserves the right** to make adjustments to students' grades based on their attendance, participation in discussions, progress, etc.

  - The instructor **reserves the right** to use the +/- grading system.

- *Course Policies & Procedures*

  - All of your submission on Canvas should be sorted in the order of questions and submitted as a singled PDF file. If your solutions are hand-written they must be scanned using a scanner app. Otherwise, they will not be considered.

  - *Quizzes*

    * Short quizzes will be given throughout the course at the beginning of some classes.
    * Often, each quiz includes one of the lectures listed below, and it will be given after the last class we finish covering a lecture (not necessarily exactly right after the last class).
    * Requests for make-up will not be entertained.

  - *Exams*

    * Requests for make-up examinations will not be entertained. I will only make exceptions to this policy in case of excused absences. You must provide sufficient documentation to prove that your absence is excused.
    * The date provided for the exam is tentative. Changes to an exam date will be announced in-class at least one week prior to the exam.
    * For online exams, you will be asked to turn on your camera during the examination. If you have any difficulty doing so, you should let the instructor know within the first week of classes.
    * The final exam is cumulative.

  - *Homework Assignments*

    * For each lecture, a homework assignment with a set of solutions will be uploaded on Canvas.
    * The assignments will not be graded. However, you are strongly encouraged and expected to work on the homework questions like if they were going be graded!
    * You should not look at the solutions until you solve or work sufficiently on each problem (so, basically, look at the solutions just to confirm your answer to the problem or in case you could not solve the problem).
    * Start working on an assignment after we start covering the related topic.

- * Although the assignments will not be graded, you are expected to work on them as if you want to submit your solutions to be graded by a deadline (a day after the last class of each lecture).
  - * **Take the homework assignments seriously!**
- – *Regrade Policies*
  - * Regrade requests must be submitted, in writing, to the instructor **within seven calendar days** of either: (1) the date the graded material is returned in class or (2) the date the grades are posted on Canvas, whichever occurs first.
  - * Regrade requests must be written on a separate sheet of paper and must be attached to your original submission.
  - * Regrade requests must specify the question(s) or parts to be regraded. Regrade requests must include a brief description of why the question(s) or parts should be regraded.
  - * The instructor reserves the right to regrade the entire submission.
  - * Graded material which has been modified in any way since it was returned to the student will not be regraded.
- – *Attendance Policies*
  - * Students are expected to **attend all classes**.
  - * The participation grades for students are computed based on each student's attendance and also the level of participation in discussions in class.
  - * You are responsible for all material presented during each lecture. Material presented during the lectures may not be in the textbook or the uploaded material.
  - * Students who anticipate the necessity of being absent from class due to the observation of a major religious observance must provide notice of the date(s) to the instructor, in writing, by the second class meeting.
- – Hand-written work must be legible. If your work is illegible it will not be graded (i.e., you will earn a grade of 0).
- – The instructor reserves the right to interpret the class policies if confusions occur.

- *Academic Integrity/Academic Dishonesty*

  Computer Science students are required to adhere to campus and departmental guidelines for academic integrity. Campus guidelines are outlined in the CSUS University Policy Manual on Academic Honesty (https://www.csus.edu/umanual/student/stu-0100.htm). The following is additional information specific to Computer Science courses.

  Avoid Cheating!

  - – *Avoid excessive collaboration on individual assignments.* A healthy amount of discussion between students is a valuable part of learning. But it is not acceptable to collaborate in developing the details of a solution, unless the assignment is a team project.
  - – *Do not give answers to other students.* Giving fellow students answers they were supposed to determine on their own is cheating. Doing so also deprives them of the learning experience of reaching a solution through their own efforts.
  - – *Keep all printouts, diskettes, notes, etc. secure*, so that other students will not find them and use your solutions in their work. This also includes shared network drives, printer queues, temporary directories in lab workstations, etc.
  - – *Limit discussion to high-level concepts.* Discussing concepts, assignment requirements, syntax errors, coding tricks, or programming environments is generally encouraged. When discussion involves specific code or solutions, it may cross the line into cheating.
  - – *Always stop and think before copying or emailing any source code.*
  - – *Ask your instructor if you are unsure.* Students who are unsure whether their activities might be considered cheating are encouraged to consult with their instructor. Openness can make a cheating instance considerably less flagrant than if it were covered-up.

The CSU is committed to providing an electronic environment that is accessible to everyone, including individuals with disabilities. If you encounter any accessibility problems and need assistance, please contact Services to Students with Disabilities: `http://www.csus.edu/sswd/`; `sswd@csus.edu`; (916) 278-6955; (916) 278-7239 (TTY).

- *Tentative Schedule*

    - Lecture 1: Asymptotic Analysis (3 meetings)
    - Lecture 2: Recursion and Backtracking (4 meetings)
    - Lecture 3: Divide and Conquer (4 meetings)
    - Lecture 4: Solving Recurrences (3 meetings)
    - Lecture 5: Graph Basics (1 meeting)
    - Lecture 6: Greedy Algorithms (2 meetings)
    - Lecture 7: Dynamic Programming (4 meetings)
    - Lecture 8: Network Flow (2 meetings)
    - Lecture 9: NP-Completeness (3 meetings)

- *Exams*

    - Midterm: Lectures 1–4
    - Final Exam: Cumulative

Monday	Tuesday	Wednesday	Thursday	Friday
Aug 30th **1** Orientation	31st	Sep 1st **2** Lecture 1 (cont.)	2nd	3rd
6th **Labor Day**	7th	8th **3** Lecture 1 (cont.)	9th	10th
13th **4** Lecture 2	14th	15th **5** Lecture 2 (cont.)	16th	17th
20th **6** Lecture 2 (cont.)	21st	22nd **7** Lecture 3	23rd	24th
27th **8** Lecture 3 (cont.)	28th	29th **9** Lecture 3 (cont.)	30th	Oct 1st
4th **10** Lecture 4	5th	6th **11** Lecture 4 (cont.)	7th	8th
11th **Indigenous People's Day**	12th	13th **12** Lecture 4 (cont.)	14th	15th
18th **13** Midterm	19th	20th **14** Lecture 5	21st	22nd
25th **15** Lecture 6	26th	27th **16** Lecture 6 (cont.)	28th	29th

Monday	Tuesday	Wednesday	Thursday	Friday
Nov 1st **17** Lecture 7	2nd	3rd **18** Lecture 7 (cont.)	4th	5th
8th **19** Lecture 7 (cont.)	9th	10th **20** Lecture 7 (cont.)	11th ***Veteran's Day***	12th
15th **21** Lecture 8	16th	17th **22** Lecture 9	18th	19th
22nd **No Class**	23rd	24th ***Thanksgiving Break***	25th	26th ***Thanksgiving Break***
29th **23** Lecture 9 (cont.)	30th	Dec 1st **24** Lecture 9 (cont.)	2nd	3rd
6th **25**	7th	8th **26**	9th	10th