Syllabus for CECS 528, Advanced Algorithms

Instructor: Dr. Todd Ebert

Fall 2026, Last Updated August 7th, 2026

General Course Information

Academic Unit Department of Computer Engineering and Computer Science, California State University, Long Beach

Prerequisite Open to Computer Science MS, Computer Engineering MS, or Engineering MS students only, or consent of instructor. Students are strongly encouraged to have completed an undergraduate course in data structures and algorithms similar to that of CECS 328.

Catalog Description Theoretical analysis of algorithms. Divide and conquer, dynamic programming and greedy algorithms; basic search and traversal techniques including search trees; sorting; matrix manipulations; NP-completeness.

Course Meeting Times MW 8:00-9:15 am (Section 1, 3664), TuTh 8:00-9:15 am (Section 2, 10910)

Instructor Dr. Todd Ebert (Todd.Ebert at csulb.edu)

Instructor Office Hours MW 10:45-11:15, TuTh 9:30 am - 11:00 am, ECS 548

Textbook S. Dasgupta, C. Papadimitriou, U. Vazirani, **Algorithms**, McGraw-Hill Education, 1st edition, 2006

Course Topics

- Review of Big-O notation
- Analysis Tools for Divide-and-Conquer Algorithms: Recurrences, Master Theorem, Substitution Method
- Divide-and-Conquer Algorithms: Mergesort, Quicksort, Finding Array Statistics, Finding the Median of an Array, Karatsuba's Integer Multiplication Algorithm, Strassen't Matrix Multiplication Algorithm, Maximum Subsequence Sum, Minimum Positive Subsequence Sum, Fast Fourier Transform

- Data Structures for Greedy Algorithms: Binary Heaps, Disjoint Set Data Structure
- Greedy Algorithms: Dijkstra's, Prim, and Kruskal's algorithms, Unit Task Scheduling, Task Selection, Fractional Knapsack, Fuel Reloading
- Replacement Method for proving correctness of greedy algorithms
- Dynamic Programming Algorithms: 0-1 Knapsack, Edit Distance, Optimal Binary Search Tree, Longest Common Subsequence, Minimum and Maximum Single Source Distances in DAGs, Floyd Warshall Algorithm, Runaway Traveling Salesperson.
- Logic Problems: 2SAT, 3SAT, XOR-SAT, and Horn-SAT
- Mapping Reducibility: Even to Odd (and vice versa), Independent Set to Clique (and vice versa), Embeddings, Set Partition to Subset Sum, Hamilton Path to LPath, Contractions, Subset Sum to Set Partition, Vertex Cover to Half Cover, Clique to Half Clique, Max Matching to Max Flow
- Introduction to Computational Complexity Theory: complexity classes `P`, `NP`, and `co-NP`, `NP`-complete problems, Cook's Theorem, mapping reductions that establish NP-completeness, SAT to 3SAT, 3SAT to Clique, 3SAT to Subset Sum, 3SAT to Directed Hamilton Path (DHP), Independent Set to Vertex Cover, DHP to Hamilton Path, Hamilton Path to Hamilton Cycle, Hamilton Cycle to Traveling Salesperson
- Approximation algorithms: Vertex Cover, Traveling Salesperson, Data Clustering, Load Balancing, Max Cut
- Review of Probability Theory: random variables and events, expectation and variance of a random variable, linearity of expectation, Independence of random variables and events, Bernoulli, Geometric, and Binomial random variables, conditional probability, conditional expectation, conditional-expectation random variables, chain rule (including generalized version).
- Randomized algorithms: Min Cut, 3SAT, Randomized Quicksort, Randomized Finding Array Statistics, Max Cut, Hashing, Set Balancing

Learning Outcomes

This course has twelve core **learning outcomes (LO's)** that will be regularly assessed throughout the semester. For each outcome, your understanding is assessed via a problem that is provided on an **assessment**, also commonly referred to as a *quiz* or *exam*. On your first attempt at demonstrating competency with respect to a learning outcome, you earn both points as well as a pass (P) or no pass (NP) grade. Receiving a pass grade means that you have successfully demonstrated competency with respect to that LO and are no longer required to demonstrate competency.

The following are some guidelines for success towards passing each of the learning outcomes.

1. When preparing for the learning-outcome problem, carefully read its official description in the syllabus so that you know what is expected for a passing mark.

2. Each LO has a set of core exercises (with solutions) to help you prepare. These exercises are located at the end of the lecture that introduces the LO.

3. During assessments, you are allowed to bring pens/pencils, eraser, and a non-programmable scientfic calculator **All other computing devices (cell phones, laptops, smart watches, etc.), notes, and books must be put away before starting. Before receiving the assessment you should have your space cleared and allow me to witness you putting your phone and watch away in a closed compartment. Failure to abide by these rules is grounds for receiving a final course grade of F without the possibility of having the grade forgiven**.

4. To help prevent cheating during in-class assessments, whenever possible, please select a seat that leaves one or more empty seats between you and your nearest neighbor. **When there is sufficient evidence of student plagiarism during an assessment, all involved students are subject to receiving a final course grade of F without the possibility of having the grade forgiven**.

5. Carefully read the directions for each problem and include all the pertinent steps of your solution. The problem statement is like a contract in that, so long as you provide exactly what is being asked of you and show sufficient work, then the outcome should be a passing mark. Please do not hesitate to raise your hand if you have any doubts about the contract is requiring of you.

6. Solve each LO assessment problem on a **SINGLE and SEPARATE** sheet of paper that includes your first name and surname (the one that determines your alphabetical order).

7. During assessments, both a five and two-minute warning will be announced before solutions are to be turned in. Please have the courtesy to turn in your work before the final call. A student who is still working after the final call will receive a warning and any subsequent infractions will result in the student's work not being graded.

8. At the end of the semester each student will receive grade points in accordance with the following table and based on the number of LO's passed (LO's 1-10 only).

Number Passed	**Grade Points**
0-3	0
4	0.5
5	1.0
6	1.5
7	2.0
8	3.0
9	3.5
10	4.0

The following are the LO's for which competency must be demonstrated.

LO1. The ability to use the Master Theorem to determine the growth of a function that satisfies a uniform divide-and-conquer recurrence, and the ability to use the Substitution Method to provide either a big-O upper bound or big-Omega lower bound for the growth of some function that satisfies a some divide-and-conquer recurrence.

LO2. An understanding of the concepts that establish the correctness or running time of a divide-and-conquer algorithm, and the ability to demonstrate the steps of the algorithm. The list of algorithms are Mergesort, Quicksort, Find-Statistic, Strassen's algorithm, Karatsuba's Integer Multiplication algorithm, Maximum Subsequence Sum, and Minimum Positive Subsequence Sum.

LO3. The ability to demonstrate the steps of a greedy algorithm, including Dijkstra's, Prim's, and Kruskal's algorithms, Unit Task Scheduling, Task Selection Algorithm (TSA), Fractional Knapsack, and Fuel Reloading. An understanding of how the binary heap and disjoint-set data structures are used for improving the running time of some greedy algorithms (Unit Task Scheduling, Kruskal's, Prim's, and Dijkstra's algorithms) . The ability to demonstrate the use of one of these data structures within the context of Kruskal's, Prim's, Dijkstra's, and the `Unit Task Scheduling` algorithm.

LO4. The ability to provide a dynamic-programming recurrence and demonstrate the corresponding dynamic-programming algorithm for some problem studied either in lecture, or in the lecture exercises, including 0-1 Knapsack, Edit Distance, Longest Common Subsequence, Optimal Binary Search Tree, Matrix-Chain Multiplication, Shortest and Longest Paths in an Acyclic Graph, Runaway Traveling Salesperson, and the Floyd-Warshall algorithm.

LO5. The ability to demonstrate the steps of the improved `2SAT` algorithm and understand the conceptual underpinnings of the algorithm in terms of the interrelationship between logical properties and graph properties of the associated implication graph.

LO6. The ability to provide the definition of what it means to be a mapping reduction f from some problem A to another problem B, and demonstrate one of the following reductions, and verify that the solution to $x \in A$ the same as the solution to $f(x) \in B$.

(a) `Even` $\leq_m$ `Odd` See Exercise 1 of the Mapping Reducibility lecture for the kind of problem to expect.

(b) `Max Independent Set` $\leq_m$ `Max Clique` and vice versa

(c) `Set Partition` $\leq_m^p$ `Subset Sum`

(d) `Hamilton Path` $\leq_m^p$ `LPath`

(e) `Subset Sum` $\leq_m^p$ `Set Partition`

(f) `Vertex Cover` $\leq_m^p$ `Half Vertex Cover`

(g) `Clique` $\leq_m^p$ `Half Clique`

(h) `Max Bipartite Matching` $\leq_m^p$ `Max Flow` (If time permits)

LO7. An understanding of the complexity classes `P`, `NP`, and `co-NP` and the kinds of problems that belong in each class. Also, given a description of some NP decision problem L along with its certificate set, the ability to provide pseudocode for its verifier that accepts the input instance x assuming the input certificate is valid for x. Also the ability to analyze the big-O number of steps of the verifier to establish that it is bounded by a polynomial.

LO8. The ability to demonstrate and/or answer questions about one of the following polynomial-time mapping reductions that establishes the NP-completeness of some decision problem based on a reduction from 3SAT: 3SAT to Clique, 3SAT to Subset Sum, and 3SAT to DHP. Also, the ability to demonstrate one of the following additioinal reductions: SAT to 3SAT, DHP to UHP, and Hamilton Cycle to Traveling Salesperson.

LO9. The ability to demonstrate the steps of an approximation algorithm on some problem instance and understand why a certain approximation algorithm achieves a certain approximation ratio. Algorithms include those for Min Vertex Cover, k-Clustering data, Load Balancing, and Traveling Salesperson.

LO10. The ability to demonstrate a randomized algorithm for one of the following problems: Find Statistic, Min Cut, Matrix Multiplication Verification. Also, the ability to answer a probability question in relation to one of the following randomized algorithms: Find Statistic, Quicksort, Min Cut, Matrix Multiplication Verification, and Walk SAT.

LO11. **Working General Knowledge.** A working knowledge of all the basic definitions and concepts introduced in the class. In other words, being able to recite the definitions and results and use them within the context of basic logic and problem solving.

LO12. **Advanced Ability.** The ability to apply and extend the core skills acquired from previous learning outcomes to solve more advanced problems.

Weekly Exercises

Associated with each learning outcome presented in lecture will be a set of assigned practice exercises, as well as some additional exercises that pertain to a related peripheral learning outcome and/or represent a more advanced exercise that may form the basis for an Advanced Ability assessment problem. Working General Knowledge problems are based on definitions and results taken directly from the lecture notes.

Class Meetings

Our class meetings will be devoted to the following three activities.

Course lecture notes These notes have several examples. Some have provided solutions while the solutions to others will be demonstrated in class. The notes also contain all the needed definitions, formulas, theorems, exercises, and exercise solutions that are needed to perform well in the class.

In-Class Assignments (ICA's) For some class meetings time will be alloted for students to work in groups of three to solve a collective problem that reinforces understanding of one of the course learning outcomes. Work on ICA's counts towards 10% of a student's final grade. A student must be present in class to receive credit for an ICA and all ICA assignment papers must be turned in by the end of the class meeting for which it was assigned. Students should expect to have an ICA at every class meeting, except for the first week of class and the meeting of, the meeting before, and the meeting after a learning outcome assessment. Note: there are **no makeups for ICA's**.

Learning Outcome Assessment (LOA) About once every three weeks (plus the final exam) there will be a learning outcome assessment. The assessment will have one problem for each of the learning outcomes that have been covered up to and including the previous week. Each assessment will feature one 25-point problem for assessing each of the following: i) working general knowledge (LO11), ii) advanced ability (LO12), and iii) each of the first ten learning outcomes that has been covered in lecture but *not previously assessed*. There will also be one "make up" problem for each of the previously assessed learning outcomes that have yet to be passed. For example, suppose LO's 1 and 2 were assessed on the first assessment, and LO's 3 and 4 are first assessed on the second assessment. Then the second assessment will have four 25-point problems (LO's 3 and 4, General Working Knoweldge, Advanced Ability) and two makeup problems (LO's 1 and 2). Students will have the entire class period to complete each LOA. Important: both Advanced Ability and Working General Knowledge problems are based on knowledge obtained *after* the previous learning outcome assessment. The final exam will have one extra cumulative General Working Knowlege problem worth 50 points. The fifth assessment will not have any Advanced Ability or General Working Knowledge problems.

LOA Dates (MW Class): September 16th, October 7th, October 28th, November 18th, December 9th, December 14th (8:00-10:00 am)

LOA Dates (TuTh Class): September 17th, October 8th, October 29th, November 19th, December 10th, December 17th (8:00-10:00 am)

Learning Outcome Assessment Makeups

Since there are a total of six (including final exam) assessments, **there are no makeups for LOA's**. However, if a student has to miss a single LOA (excluding the final exam) due to a documented accident, emergency, or illness, any missed points will be added at the end of the semester, and the zero-point totals will be replaced with values that reflect the student's overall performance during the semester. For example, if during the semester the student attained one half of the total Advanced Ability points, then the zero will be replaced with 12.5 (i.e. one half of 25 points). Acceptable documentation includes a note or letter from a medical establishment that explains the emergency or illness, a police report, and/or insurance documentation. Please note that **this is one-time forgiveness and no such accomodation will be made for missing subsequent assessments**.

Supplemental Lectures

Because there will be in-class activities throughout the semester, in order to keep pace with the course objectives, a supplemental lecture recording may be provided at the end of the week. The lecture will last anywhere between 30 to 60 minutes. It's important to review the lecture before the next class meeting, as the following in-class assignment may depend on it.

Reading Assignments

A reading assignment will be provided on most weeks of the semester. Reading the textbook will offer a somewhat alternative and more comprehensive viewpoint of the subject matter. Please check the "Reading Assignments" link at the course website for the current and past assignments. The reading assignment topics pertain to those topics that will be covered in the class meetings for the following week.

Final Grade Determination

At the end of the semester the number of LO's 1-10 that a student passed will be converted to grade points (see section on learning outcomes). Also, each of the total points earned for ICA's, General Working Knowledge, Advanced Ability, and LO's 1-10 will be converted to separate grade points. These five grade points are then weighted as follows in order to determine your final course grade points.

Category	**Percentage Weight**
Learning Outcomes Passed	30%
Learning Outcome Total Points	25%
General Working Knowledge	20%
Advanced Ability	15%
In-Class Assignments	10%

The grade points earned for each category are weighted according to the percentage weight to obtain the overall course grade point average (OGPA)

$$\mathrm{OGPA} = 0.3 \times \mathrm{LOP} + 0.25 \times \mathrm{LOTP} + 0.2 \times \mathrm{GWK} + 0.15 \times \mathrm{AA} + 0.1 \times \mathrm{ICA}$$

which is rounded to the nearest tenth. The grade point average is then converted to a letter grade: A (3.5 and above), B (2.5 to 3.4), C (1.8 to 2.4), D (0.5 to 1.7), and F (0-0.5).

Registration Deadlines

September 7th Last day to add or drop classes without approval

September 21st Deadline to add a course

November 20th Withdrawal deadline