

Syllabus of CISP300

2010 Fall/Spring Section 12776

Class information

- Class code 12776
- Course Title: Algorithm Design/Problem Solving
- Course Description: This course introduces methods for solving typical computer problems through algorithm design. Topics include assessing and analyzing computer problems in a top-down, divide-and-conquer approach that leads to a programming solution. It also covers programming plans and detailed design documents from which source code versions of programs are created.
- Student Learning Outcome:
 - define operators, including arithmetic, comparison, and logical operators.
 - differentiate control structures, including branches (conditional statements) and loops (pre-checking and post-checking loops).
 - deduce post condition from pre condition for control structures, including assignment statements, branches, and loops.
 - construct a trace table to emulate the execution of a program that utilizes variables, various control structures, data organizations, subroutines, and parameters.
 - contrast the lifespan limits and behaviors of local variables, by-value parameters, and by-reference parameters.
 - compare the two methods of passing results: by-reference parameters and return value.
 - compare in-line copy-and-paste coding with structured subroutines in terms of maintainability, defect containment, testability, and other metrics.
 - synthesize a subroutine to abstract one or more similar blocks of in-line code using local variables, parameters, and return values.
 - differentiate roles involved in software development, including developers, analysts, and test engineers.
- Time and place: MW 17:30-18:50 Room Liberal Arts 121
- Number of units: 3
- Lecture hours: 54
- Lab hours: 0
- Final exam: 12/13/2010 1745-1945

- Additional information: check the Moodle course site at <http://www.someprofs.org/moodle> or <https://www.someprofs.org/moodle> (the https link will warn about a certificate not signed by a CA)

Professor information

- Name: Tak [Auyeung](#)
- Office: Rm 7 of Liberal Arts 133
- Phone number: 916-484-8250
- Email: auyeunt@arc.losrios.edu or tauyeung@drtak.org
- Office hours: Mon-Thur: 1200-1300, Fri: 1105-1205

Behavioral expectations

- Universal
 - No disruptive behavior is tolerated
 - No (academic) dishonesty is tolerated
 - What is academic dishonesty?
 - See [this link](#) for a more complete explanation
 - What happens when it occurs?
 - The occurrence will be documented
 - The documentation will be sent to the discipline officer
 - The discipline officer will determine the appropriate action in addition to the following:
 - The involved submission (assignment, quiz, exam or etc.) will receive a maximum score of 0 (zero) points.
 - A discovery of academic dishonesty may trigger re-investigations of prior submitted work. Any prior work newly discovered/confirmed as results of academic dishonesty will be retroactively processed. This means points of such work will be deducted.
- In class (face-to-face)
 - No phone, no drink, no food and no kid
 - Raise hand and wait for acknowledgment before asking and answering questions
- Attendance
 - R-2222: I will drop students who miss the first class session.
 - R-2222: I will drop students with 6% or more unexcused absence.
 - R-2222: I will drop students who do not attend *all* of the first two (for classes that

- meet once per week) or three (for classes that meet more than once per week) class sessions.
- I am only required to accept *verified* military duty, jury duty and medical reasons as excused absences. All other absences may or may not be excused at my discretion.
- The campus health center can verify medical excuses, and it is free.
- Online
 - All students are expected to check email at least once per day
 - Email should be sent with the following information:
 - Subject line
 - Course name (e.g., CISP300)
 - Meeting days (e.g., TuTh, online)
 - Nature (e.g., “due date of assignment 4”)
 - Body
 - Details of the question/comment
 - Actual name of student (as registered)

Resources

- iMail (<https://imail.losrios.edu>) is the official point of contact for both face-to-face and online classes.
- [Moodle at someprofs.org](https://moodle.at.someprofs.org) is the course management tool for both face-to-face and online classes.

Grading

- No make up submitted work unless it is excused.
- *Definitely* no make up work once the solution is disclosed.
- Letter grades
 - < 12.5%: F
 - ≥ 12.5% and < 37.5%: D
 - ≥ 37.5% and < 62.5%: C
 - ≥ 62.5% and < 87.5%: B
 - ≥ 87.5% : A
- Categories and weights
 - Homework: 20%

- First exam: 20%
- Second exam: 20%
- Final exam: 40%

Schedule

Topic	begin date
Introduction to algorithms. Explain the role of algorithms in computer science and programming.	08/23/10
Present types of statements: sequences, conditional statements, loops. Represent control statements as pseudo code as well as graphical form. Nesting statements. Discuss basic properties of each type of statement.	08/23/10
Discuss the use of variables in an algorithm. Present methods to track variables during the execution of an algorithm.	08/30/10
Logical expressions and how they are used in various kinds of statements. The difference between a condition and a statement. Identify the pre and post conditions of a statement.	09/08/10
Top-down design: reasons and techniques. Relate top-down design to control structure selection.	09/15/10
First exam	09/22/10
Introduction to arrays and array indexing. Explain the limitations of the lack of arrays.	09/27/10
Basic algorithms involving arrays, such as searching in an unsorted and sorted array.	10/04/10
Rationale of subroutines. Kinds of parameters and local variables. The invocation of subroutines.	10/11/10
Records and user defined types. Nested aggregate types.	10/20/10
Second exam	10/25/10
Abstract data type: rationale, example and definitions.	10/27/10
Limitations of ADT. Introduction to object orientation concepts. Classes and objects.	11/03/10
Inheritance and extension. Abstract classes.	11/10/10
Complexity of algorithms. The big-O notation. Estimate of execution time.	11/22/10
File operations and algorithms that work with files, such as merge sort.	11/29/10
Syntax of a language, and how syntax is described by a meta language, such as BNF.	12/06/10