

CS 2401 Elementary Data Structures and Algorithms Spring 2007 Syllabus

1. General Information

Instructors:

- [Olac Fuentes](mailto:ofuentes@utep.edu), ofuentes@utep.edu
Office hours: Monday to Friday 11:00-12:00am, or by appointment, in CSB 208.
- [Thamar Solorio](mailto:tsolorio@utep.edu), tsolorio@utep.edu
Office hours: TR 10:30-11:30, F 10:00-11:00 or by appointment, in CSB 224-B.

Teaching Assistants (TAs):

- Luis Garcia, garcia.luisalejandro@gmail.com
Office hours: MW 11:30 - 12:30, R 2:00 - 3:00 or by appointment, in CSB 122
- Sheline John, shelinej@gmail.com
Office hours: TR 4:30 – 5:30 pm in CSB 304, M 7:00 – 9:00 pm in CSB 300
or by appointment

Peer Leaders (PLs):

- Miguel Cuellar, macuellar@utep.edu
- Steven Madrid, semadrid@utep.edu
- Marcus Valenzuela, mvalenzuela2@utep.edu
- Xochitlcualtzin Rodarte, xmrodarte@utep.edu

Lecture sections:

- MWF 12:30-13:20, Room CS 308. Instructor: Olac Fuentes
- TR 9:00-10:20, Room CS 322. Instructor: Thamar Solorio

Labs

- MWF 7:30-8:20, Room CS 301, TA: Sheline John
- MWF 10:30-11:20, Room CS 301, TA: Luis Garcia
- TR 12:00-13:20, Room CS 300B, TA: Sheline John
- TR 13:30-14:50, Room CS 301, TA: Luis Garcia

Class web site:

http://www.cs.utep.edu/tsolorio/cs2401_Sp07.html

2. Objectives and Outcomes

This is the second course for students majoring in Computer Science. Students will learn about fundamental computing algorithms, including searching and sorting; elementary abstract data types including linked lists, stacks, queues and trees; and elementary algorithm analysis.

Level 3 Outcomes: Synthesis and Evaluation:

Level 3 outcomes are those in which the student can apply the material in new situations. This is the highest level of mastery. On successful completion of this course, students will be able to identify, implement and use the following data structures as appropriate for a given problem:

- Multi-dimensional arrays.
- Lists implemented as arrays or linked lists.
- Stacks.
- Queues.
- Binary trees and binary search trees.
- Simple hashes

Level 2: Application and Analysis:

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details. Upon successful completion of this course, students will be able to:

- use Big-O notation to express the best-, average- and worst-case behaviors of an algorithm
- explain the structure and use of activation records
- determine the best, average and worst-case behaviors of an algorithm
- assess time and space trade-offs in algorithms.
- explain, code, and use quadratic and $O(n \log n)$ sorting algorithms
- implement recursive algorithms over natural numbers, lists, and trees
- define and use classes, subclasses and inheritance.
- implement a simple graphical user interface
- perform string manipulation and simple parsing
- implement and use multidimensional arrays
- describe the importance of encapsulation and information hiding
- implement applications and simulations of the data structures identified above.
- implement simple sequential and binary search algorithms
- implement quadratic sorting algorithms
- describe memory allocation of integers, real numbers, arrays and objects.

Level 1: Knowledge and Comprehension

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. On successful completion of this course, students will be able to:

- explain basic and introductory-level notions of a virtual machine

- explain the concept of polymorphism
- use class browsers and related tools
- identify class hierarchies
- recognize the basic terms associated with particular data structures e.g. head/tail, push/pop/peek

3. Policies and Other Information

Cellular telephones are prohibited during lecture and lab sessions. Students are required to turn off their cellular telephones before entering the classroom or laboratory session. If your cell phone rings while you are at a lecture or laboratory session the TA will mark you as absent.

Prerequisites: Minimum "C" grade in CS1401 or equivalent. Students are assumed to be comfortable programming in Java. Students should be able to code basic arithmetic expressions, define simple classes, use strings, code loops and conditional statements, write methods, create objects from classes, invoke methods on an object, perform basic text file input and output, and use arrays.

Attendance Policy: Students are allowed at most five absences. Three tardies will count as one absence. A tardy will be recorded each time a student shows up ten minutes after the start of class.

Textbook: Reading and laboratory assignments will be drawn from *Java™ Programming, Program Design Including Data Structures* by D.S. Malik. You are required to obtain this book for use in this course. Note that photocopied textbooks are a violation of copyright law. Any student caught with a photocopied book will be referred to the Dean of Students for discipline.

Grading: Final grades will be based on a combination of lab projects, homework assignments, in-class attendance and performance, three partial exams, and a final exam. The approximate percentages are as follows:

- 5% - Attendance to lectures, labs and PLTL sessions
- 20% - Lab projects
- 10% - Written homework assignments
- 10% - In-class exercises, quizzes, and anti-quizzes
- 30% - Partial Exams (3 exams, 10% each)
- 25% - Final Comprehensive Exam

The nominal percentage-score-to-letter-grade conversion is as follows:

- 90% or higher is an A
- 80-89% is a B
- 70-79% is a C
- 60-69% is a D
- below 60% is an F

Additionally, any one of the following **will result on a final grade of F**, even if the overall average is greater than 60%.

- Obtaining an average of less than 60% on the lab projects

- Missing more than four lab sessions
- Missing more than five lectures

We reserve the right to adjust these criteria downward, e.g., so that 88% or higher represents an A, based on overall class performance. The criteria will not be adjusted upward, however. You must earn a C or better to continue to the next course in this sequence.

Late homework submission: Homework up to a day late will receive up to 80% of full credit, and it will not be accepted after that.

Disabilities: If you feel that you may have a disability that requires accommodation, contact the Disabled Student Services Office at 747-5184, go to Room 106E Union, or email dss@utep.edu

4. Important Advising Notes:

- Students are required to register for both lecture and laboratory during the same semester. **Attendance to both is mandatory.**
- CS 2402 has two prerequisites: CS2401 and MATH 2300 (Discrete Math). If you are not currently taking MATH 2300 and have not already passed it, you should seriously consider adjusting your schedule.

5. Lab Submission Guidelines

Soft-copy submission: Working programs must be submitted online to both your TA and instructor, use the e-mail addresses listed below. Labs not submitted this way will not be eligible for credit.

- Olac Fuentes: olacfuentes@gmail.com
- Tamar Solorio: tamar.solorio@gmail.com
- Luis Garcia: utep.cs2401@utep.edu
- Sheline John: shelinej@gmail.com

Demo sessions: In addition, you must schedule a one-on-one session with your TA in which you will explain how your program works and he/she will ask questions to test your understanding of the program being submitted. The TA will then assign a pass/fail grade for this session; a student receiving a failing grade in this session will receive a grade of zero for the project. **NOTE:** The lab demo should be presented before or on the deadline for the assignment.

Late Projects: Lab projects up to a week late will receive up to 70% percent of full credit, from one to two weeks late, up to 50%, and more than two weeks late will receive no credit. **NOTE:** The turn-in-date for labs is not when you e-mail the code, but the date when you demo the lab.

Collaboration: Discussion of homework and projects among students is encouraged, but your answers and your code should be written and tested by you alone. **Do not exchange programs or let someone look at your code, even "just so they can see how you did it."** If you need help, consult one of the professors, your TA, or your peer-leader.

Laboratory Sessions: Laboratory sessions are designed to give you guidance in getting your homework assignment started well. In a typical lab session, the Teaching Assistant will present additional material that will help you complete the assignment and answer your questions as you begin working.

Once you have completed the current assignment and you have submitted it via e-mail you may work on the next assignment or leave the laboratory, if it is not a Peer-Led Team Learning session. However, during lab sessions you should not use the computers for something other than working on your lab projects. The TA will mark as absent any student violating this policy.

6. Peer-Led Team Learning (PLTL) Sessions

PLTL sessions are designed to help you practice course concepts by engaging you in group activities. These sessions will take place once a week during your regular lab sessions and will last for 50 minutes. Attendance to these sessions is mandatory and is considered as part of the 5% credit towards your final grade.

7. Standards of Conduct and Academic Dishonesty

You are expected to conduct yourself in a professional and courteous manner, as prescribed by the UTEP Standards of Conduct:

<http://studentaffairs.utep.edu/Default.aspx?tabid=4386>

Academic dishonesty includes but is not limited to cheating, plagiarism and collusion. Cheating may involve copying from or providing information to another student, possessing unauthorized materials during a test, or falsifying data (for example program outputs) in laboratory reports. Plagiarism occurs when someone represents the work or ideas of another person as his/her own. Collusion involves collaborating with another person to commit an academically dishonest act.

Professors are required to - and will - report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.