CS 1401 Introduction to Computer Science
Fall 2013 Syllabus

Class Meetings:

- lecture session MW 9:00-10:20 am, in room CCSB G.0208; final exam Wednesday, December 10, 10:00 am - 12:45 pm
- lecture session TR 9:00-10:20 am, in room NURS 221; final exam Tuesday, December 9, 10:00 am - 12:45 pm

You should be enrolled in one lab section. You must attend the lecture section and the lab section that you are enrolled in; do not drop in on another lab section without prior approval.

Instructor: Vladik Kreinovich, vladik@utep.edu, office phone (915) 747-6951
Office hours: Mondays and Tuesdays 8-9 am, Wednesdays and Thursdays 10:30-11:30 am, or by appointment,
in CCSB 3.0404.

Teaching Assistants (TAs): TBD

Peer Leaders: TBD

Course Objectives: First course for students majoring in Computer Science. Introduction to problem solving with computers, including representation, control structures, and software development methods; closed laboratory and programming assignments in a high-level language; programming environments; social and ethical aspects of computing.

Prerequisite: MATH 1508 or MATH 1411 with a grade of C or better.

Knowledge and Abilities Required Before the Students Enter the Course: Students entering the course are not required to have background in Computer Science or programming. They should be familiar with running software applications and using a computer, and they should be familiar with topics from Pre-Calculus including algebraic functions and proofs, and base representations of numbers.

Learning Outcomes

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:

1. use the syntax and semantics of a higher-level language to express their solutions to programming problems, including the correct use of:
    a. basic variable types such as integer, real number, character, string, 1-D array
    b. assignment, arithmetic, and logical operations
    c. basic control structures: if-then, for-loop, while-loop

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:

1. use the syntax and semantics of a higher-level language to express their solutions to programming problems, including the correct use of:
    a. complex arithmetic and logical expressions
    b. simple I/O operations
    c. methods, i.e., user-defined subprograms, including recursive methods
    d. classes, i.e., user-defined types
2. describe computer representation of simple data types and operations, including operations with binary numbers
3. use testing and debugging strategies, including black-box and white-box testing, test drivers, stubs and test suites, to identify software faults
4. use exceptions to handle violation of preconditions
5. use teamwork roles and methods in the classroom

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:

1. describe, at a high level:
    a. technical aspects of computing: memory, operating system, editors, interpreters, compilers, debuggers, virtual machine
    b. computing as a profession, from required knowledge and skills to major career options
    c. relation between computing and society, including main social, ethical, and legal issues
    d. history of computing

Textbook: Reading and laboratory assignment concepts for the Java Programming Language will be drawn from *Introduction to Java Programming: Comprehensive Version*, 10th Ed., by Y. Daniel Liang (Prentice Hall, 2015). It is Ok if you have already purchased the 9th edition. You are expected to acquire a copy for your use in this course. Photocopied textbooks are illegal and their use will not be tolerated.

P.S. If you have already tried to take this class in the past, and, as a requirement for this class, you have already purchased the 9th edition, it is OK to use the 9th edition for this class as well. Please note, however, that some section numbers are different in the 10th edition.

Software: Software used in this course will be available on the Windows computers in the main computer lab and in the two instructional labs on the first floor. For those who wish to use the course software on your home computer, instructions will be given in the labs.

Assignments: Reading and homework assignments will be handed out, announced in class and in labs, and/or posted on the class website http://www.cs.utep.edu/vladik/cs1401.14. Home assignments are due by the beginning of the class on the due date. If you miss a class or lab session, it is your responsibility to find out what you missed. You should expect to spend at least 10 hours/week outside of class and lab on reading and homework.

Grading: Your semester grade will be based on a combination of homework and lab assignments, weekly quizzes, lab attendance, exams, and a final exam. The approximate percentages are as follows:

- 26% Homework and lab assignments (approximately 12-13 lab assignments)
- 10% Lab attendance and participation
- 3% Experiment participation or short research paper (2 experiments or papers)
- 10% Quizzes (expect weekly)
- 36% Exams (3 exams)
- 15% Final exam

Each of these is explained in more detail below. 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

You must earn a C or better to continue to the next course in this sequence.

Homework and Lab Assignments: Reading assignments are intended to prepare you for the class. Homework and lab assignments are designed to allow you to practice using the concepts presented in lecture and in your reading. Homework and lab assignments may include written problems, tutorial exercises, and programming problems. Assignments usually will be due at the start of your lab session. Late homework will be accepted only in unusual circumstances, by prior arrangement if at all possible.

Homework and lab assignments must be done individually, unless otherwise instructed. For some lab assignments, you will be working with a partner on pair programming assignments (see below for more details).

Attendance: Attendance at and participation in all class and lab sessions are critical components of this course. Students should attempt to be on time for all scheduled sessions and attend the entire session. Students should notify the instructor prior to missing a session if at all possible. The instructor will allow two unexcused absences per semester before having the option to deduct points from the final grade. Any assignments due on the date of the absence will be considered late if not turned in as specified by the assignment guidelines unless an exception is granted by the instructor. Points lost due to an unexcused absence may not be made up. Any points lost due to an excused absence will need to be made up by arrangement with the instructor. It is the student's responsibility to obtain the content covered during missed class(es).

Laboratory Sessions: You are required to sign up for and attend the lab section associated with this course, meeting two times a week for a total of 3 hours per week. Laboratory sessions are designed to give you guidance and assistance with projects. In a typical lab session, a Teaching Assistant will present additional material that will help you understand the concepts and complete your assignments successfully. They also will model professional techniques and work habits, and you should feel free to ask questions. Although you may discuss lab assignments in a general way with other students, if you (or you and your partner(s) for team projects) need help coding or testing your programming assignments you should consult the TA, a tutor (in the ACES lab), or the instructor, rather than other students. **Do not exchange programs or let other students look at your code, even "just so they can see how you did it."**

Your TA will be grading lab assignments, and you should work closely with the TA to make sure you understand the assignment and any specific instructions clearly. Lab attendance is part of your grade and you should make arrangements with your TA (and your lab partner(s) on team projects) if you must miss a lab. To earn full credit for attendance, you must show up on time, stay for the entire session, and work only on your assignment. You will be working individually on lab projects most of the time, but some assignments require that you work with a lab partner or partners. For those projects, your participation and cooperation with your lab partner(s) will be a factor in your grade.

Research Participation: During the course of the semester, you must participate as a subject in two Computer Science experiments or write two short (1-2 pages each) papers describing some aspect of the research going on within the Computer Science department. The purpose of this requirement is to make you aware of the research going on within this department (or elsewhere) and/or to allow you to participate in an authentic research experience. Detailed instructions for this requirement are supplied in a separate document.

Quizzes: The purpose of each quiz is to ensure that you are staying current with the weekly reading assignments and to verify that you have mastered the major concepts of recent lectures. Quizzes typically will be about 10-15 minutes in length and may include material assigned for upcoming lectures in addition to selected concepts from previous lectures.

Quizzes usually will be given at the beginning of class. There will be no make-up on missed quizzes.

Exams: The purpose of the exams is to allow you to demonstrate mastery of course concepts. Each exam will focus on the material from the previous four or five weeks. Exams will take place during the regular class session. There will be four exams, contributing 36% to the overall course grade.

Because the exams contribute so heavily to your total grade, it is vital that you do well on them. If you have test-taking difficulties in general, or if you have difficulties with our tests in particular, please come let me know as soon as possible.

Exams are planned for the following dates (subject to change):

- Monday September 22 or Tuesday, September 23, depending on your class section (Week 5)
- Monday, October 13 or Tuesday, October 14 (Week 8)
- Wednesday, November 19 or Thursday, November 20 (Week 13)

Make-up exams will be given only in extremely unusual circumstances. If you must miss an exam, please meet with an instructor, BEFORE the exam if at all possible.

Final Exam: The final exam will be comprehensive and will count 15% toward your course grade. You must score 60% or better on the final exam to pass this course. If you are planning to take CS 2401 and cannot demonstrate your understanding of the concepts and skills needed for success in CS 2401, you cannot pass this course even if you have enough points to do so.

You must take the final exam during the time shown in the schedule for the class section that you are formally enrolled in. Do not simply "drop in" to the other section; there will not be a copy of the exam for you. This is University policy. If you have a scheduling conflict (e.g., if you are taking a final at EPCC) or if you are scheduled for three final exams in one day, see one of the instructors IN ADVANCE.

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

Graded work (for example, homework or exams), is to be completed independently and should be unmistakably your own work (or, in the case of pair work, your pair's work). You may not represent as your own work material that is transcribed or copied from another person, book, or any other source, such as a web page. Professors are required to--and will--report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.

Use of Electronic Devices during Class Sessions: Any use of electronic devices that disrupts the learning environment (e.g., surfing the Web, listening to music, checking Facebook, Twittering, playing Angry Birds Rio or online Scrabble while class is in session) will not be tolerated. Electronic devices should serve as tools for learning and are limited to course-related work only; any other use is considered inappropriate. Inappropriate use of any electronic devices will be considered a disruption of the classroom and may be reported to the Dean's office. All electronic devices should be silenced or shut off upon entering the classroom. Only when lecture has ended or when the instructor is not giving a lesson should students be allowed to use electronic devices as long as they are on task. In the event of an emergency or other urgent situation, the student should step outside of the classroom beyond hearing range or text silently. It is the student's responsibility to ensure that all electronic devices are managed within the guidelines. The instructor reserves the right to disallow use of any electronic equipment during class sessions.

Disabilities: If you feel that you may have a disability that requires accommodation, contact the Center for Accommodations and Support Services at 747-5184, go to Room 106E Union, or email cass@utep.edu

Weekly Schedule: (tentative and subject to change) When learning a new topic, often, we will see how it relates to the previously learned topics and thus, we will use this as an opportunity to recall the previous topics

1. Reading: Chapter 1, Introduction to Computers, Programs, and Java, Sections 1.1-1.8 (10th edition)
    - August 25-29: administrivia, groupwork, why CS?, groupwork, problem-solving techniques, an overview of the history of computers, elements of computer systems, algorithms, programming languages, classes and objects (a brief introduction)
2. Reading: Chapter 2, Elementary Programming, Sections 2.1-11
    - September 2-5: simple input, identifiers, variables, assignment statements, constants, data types, arithmetic operations, arithmetic expressions, operator precedence
3. Reading: Chapter 2, Elementary Programming, Sections 2.13-15, 2.17-18; also, Sections 1.9, 1.10, 4.3, and 4.4
    - September 8-12: augmented assignment operator, increment and decrement operators, type conversions, characters, strings, programming style and documentation
4. Reading: Chapter 3, Selections, Chapter 3
    - September 15-19: boolean data types, if-statements, logical operations, switch statements brief review for Exam 1
5. Reading: Chapter 3, Selections, Chapter 3
    - September 22-26: EXAM 1, Selections (cont-d)
6. Reading: Chapter 5, Loops, Sections 5.1-5.6
    - September 29 - October 3: while-loops, for-loops, nested loops
7. Reading: Chapter 5, Loops, Section 5.8-5.11

    - October 6-10: loops: case studies, brief overview for EXAM 2
8. Reading: Chapter 6, Methods; Section 2.16
    - October 13-17: EXAM 2, void methods, value returning methods, binary and hexadecimal numbers, conversion between decimal and binary numbers, black box/white box testing, constructing a test plan with reasonable test cases, testing programs using conditional statements
9. Reading: Chapter 6, Methods; Section 2.16
    - October 18-24: review of Exam 2 results, methods (cont-d)
10. Reading: Chapter 7, Single-Dimensional Arrays
    - October 28 - November 1: array basics, passing arrays to methods, searching in arrays
11. Reading: Chapter 9, Objects and Classes, and Chapter 10, Object-Oriented Thinking
    - November 3-7: defining classes, constructing objects using constructors, accessing objects, static variables, field encapsulation, passing objects to methods, arrays of objects
12. Reading: Chapter 9, Objects and Classes, and Chapter 10, Object-Oriented Thinking
    - November 10-14: scope of variables, class abstraction and encapsulation, primitive data types as objects, teamwork models and roles in the software process,
13. Reading: Chapter 14, GUI Basics
    - November 17-21: Java Graphical User Interface, frames, buttons, labels; brief overview for Exam 3, EXAM 3
14. Reading: Chapter 14, Exception Handling, Sections 14.1-14.5
    - November 24-26: Exam 3 results, exception overview, exception types, when to use exceptions, ethics in computing,
15. Reading: none
    - December 1-4: ethics in computing, review for final exam