CS 1101 Introduction to Computer Science

Spring 2019 Syllabus

Time and Location: TR 3:00pm-4:20 PM in CCSB 1.0704 **Textbook:** *Programming in Java*, by Zybooks. 1. Sign in or create an account at **learn.zybooks.com** 2. Enter zyBook code: **UTEPCS1301AkbarSpring2019** 3. Subscribe	**TA:** Angel Garcia (afgarciacontreras@miners.utep.edu) **TA Office Hours:** TBA **Location:** CCSB 1.0706 **Instructor:** Monika Akbar (makbar@utep.edu) **Office:** CCSB 3.0422 **Office Hours:** TR 3:00 – 4:00 PM or by appointment

Note: You should be enrolled in **one lab section**. **Your lab and class should have the same instructor.** Do not drop in on a lab or lecture section other than yours, without prior approval from your instructor.

NOTE: When contacting the instructor or TA by email, please use in the subject the prefix [CS1101].

Lab Objectives: Students will learn the foundations of algorithmic thinking and algorithm development, and learn how to implement them in a variety of languages. They will also learn to be active learners. They will develop problem-solving skills and build team skills, critical-thinking skills, and professionalism.

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

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

Course materials: All the course materials will be available through **Blackboard (Bb)**. Please check Bb regularly to stay updated with the lab.

Important lab rule about using your personal laptop computers

It is your choice to use your personal computer or UTEP's desktop to complete the labs assigned to you. However, it is essential that you be able to show your work anytime we ask you for it in the lab. For instance, we will not accept that *your work is on your laptop – or somewhere else – and you cannot produce it at the time we request it*. To avoid such situation, you could, for example, use Dropbox (dropbox.com on which you get extra free space based on your utep.edu address) and hence make sure that you can access your work from anywhere. Any option you pick, you need to be able to produce your work at any time in lab for our review and grading. **There will be no exception to this rule.**

GRADING

Grades are turned in to students in a timely manner. It is the students' responsibility to keep track of their grades by compiling the grades they receive. Your semester grade will be based on a combination of lab assignments, pop quizzes, lab participation, and a final exam.

Grade for 1101: The approximate percentages are as follows:

- 70% Labs assignments (including demos and comprehensive labs).
- 10% Homework (may include Zybook exercises).
- 20% Lab participation (includes on-time attendance, participation in labs, any quizzes for attendance and survey purposes)

The nominal percentage-score-to-letter-grade conversion for CS 1101 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

Note: *You must earn a C or better in each of these two courses, CS1301 and CS1101, to continue to the next course in this sequence, which is CS2401.*

EXPECTATIONS

Lab Assignments: Lab assignments are designed to allow you to practice the topics that constitute the outcomes of this course. Assignments will be a mix of:

- Problems to be solved without computers to practice problem solving and algorithm design.
- Programming assignments.

Deadlines for lab assignments will be clearly specified in the description of each assignment. Assignments turned in up to three days late will have scores reduced by 10% for each day of lateness.

When assessing labs, TAs will spend 5 to 10 minutes with each student asking probing questions about the topics covered in the assignments: these questions will be asked regardless of whether you completed the assignment or not. This allows you flexibility, in case something happened, and you were not able to complete an assignment, to make up for some points.

Comprehensive labs are comprehensive programming assignments. Typically, there would be 3 comprehensive labs. These labs require more time to complete. The deadline for comprehensive lab is usually longer than the usual labs.

Lab Participation: Attendance at and participation in all lab sessions are mandatory and critical factors of your success in this lab course.

Students should be **on time** for all scheduled sessions and **attend the entire session**. Attendance will be taken at every session and will count towards your class participation grade. Programming activities assigned by the TA will count towards lab participation.

Students should **notify the TA prior to missing a session** if at all possible, and certainly right after if earlier was not possible. The TA will allow two unexcused absences per semester before having the option to deduct points from the final grade (5 points per subsequent unexcused absence).

It is the student's responsibility to obtain the content covered during missed labs. Participation points may also include completing post-labs online quizzes (when applicable) that are administered as surveys to monitor students' overall progress and potential struggles.

RESOURCES

Special Accommodations: If you have a disability and need classroom accommodations, please contact the Center for Accommodations and Support Services (CASS) at 747-5148 or by email to cass@utep.edu, or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at www.sa.utep.edu/cass. CASS' staff are the only individuals who can validate and if need be, authorize accommodations for students with disabilities.

Scholastic Dishonesty: Any student who commits an act of scholastic dishonesty is subject to discipline. Scholastic dishonesty includes, but not limited to cheating, plagiarism, collusion, submission for credit of any work or materials that are attributable to another person.

Cheating is:

Copying from the test paper of another student
Communicating with another student during a test to be taken individually

Giving or seeking aid from another student during a test to be taken individually
Possession and/or use of unauthorized materials during tests (i.e. crib notes, class notes, books, etc.)
Substituting for another person to take a test
Falsifying research data, reports, academic work offered for credit

Plagiarism is:

Using someone's work in your assignments without the proper citations
Submitting the same paper or assignment from a different course, without direct permission of instructors

To avoid plagiarism, see: http://sa.utep.edu/osccr/wp-content/uploads/sites/8/2012/09/Avoiding-Plagiarism.pdf

Collusion is: Unauthorized collaboration with another person in preparing academic assignments

Important! When in doubt on any of the above, please contact your instructor to check if you are following authorized procedure.

DETAILED LEARNING OUTCOMES

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 describe, at a high level:

1. Computer representation of simple data types and operations, including operations with binary numbers
2. Technical aspects of computing, including memory, operating systems, editors, interpreters, compilers, debuggers, and virtual machine
3. Differences among programming languages
4. The purpose and use of exceptions
5. Pseudocode and implementation in a programming language of the use of Multi-D arrays
6. Pseudocode and implementation in a programming language of the use of Linked lists

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:

1. To analyze problems and express solution algorithms in pseudocode
2. To implement pseudocode algorithms in a high-level language, including the correct use of:
    a. Arithmetic and logical expressions
    b. Simple I/O operations
    c. User-defined subprograms, including recursive methods
    d. User-defined types
3. To use testing and debugging strategies, including black-box and white-box testing, test drivers, stubs and test suites, to identify software faults
4. Use teamwork roles and methods in the classroom

Level 3: 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 use the syntax and semantics of a higher-level language to express solutions to programming problems, including the correct use of:

1. Basic variable types such as integer, real number, character, string, 1-D array
2. Assignment, arithmetic, and logical operations
3. Basic control structures: if-then, for-loop, while-loop