COMPUTER SCIENCE I
CSCE 1030.002
FALL 2018

Course Instructor: Dr. Pradhumna Shrestha

Office: NTDP F265

E-mail Address: pradhumna.shrestha@unt.edu

- Include CSCE 1030.002 in subject line
- Always use your official UNT email address

Class Location/Time: ESSC 255, MoWe 3:30 PM – 4:50 PM

Office Hours: TuWeTh 2:00 PM –3:00 PM *or by appointment*

Required Textbook: We are using a new e-book called zyBook this semester:

1. Sign in or create an account at learn.zybooks.com
2. Enter zyBook code

   UNTCSCE1030.002ShresthaFall2018

3. Subscribe

A subscription is $65.70 and will last until December 27, 2018.

Optional Reference Textbook: Problem Solving with C++, Tenth Edition, by Walter Savitch, Textbook: Addison-Wesley, 2018. ISBN 978-0-13-444828-2.

Canvas
This course will use the Canvas learning management system (LMS) to distribute course materials, communicate and collaborate online, post grades, and submit assignments. You are responsible for checking the Canvas course site regularly for class work and announcements.

COURSE DESCRIPTION
CSCE 1030 is the introductory course for the computer science, computer engineering and information technology degrees offered by the Department of Computer Science and Engineering. As such it introduces students to the broad discipline of computing while placing emphasis on developing students' programming skills. In addition to three 50-minute "lecture" classes per week, each student will participate in a 2-hour 50-minute lab session each week.

TOPICS (subject to change)
1. Basic CPU Architecture
2. Basic Data Types
3. Program Structure and Design
4. Algorithms

5. Comments
6. Basic Compiler understanding and operation
7. Pre-processor instructions
8. Expressions, statements and operators
9. Arithmetic and logical expressions
10. Loops and conditionals and other flow control
11. Functions
12. Arrays
13. Console I/O both C style and C++ style
14. Pointers
15. File I/O both styles
16. Strings both CStrings and String class
17. Structures and unions
18. Command Line Arguments
19. Using libraries
20. Debugging

COURSE OUTCOMES

Course outcomes are measurable achievements to be accomplished by the completion of a course. These outcomes are evaluated as part of our ABET accreditation process.

1. Describe how a computer's CPU, Main Memory, Secondary Storage and I/O work together to execute a computer program.
2. Make use of a computer system's hardware, editor(s), operating system, system software and network to build computer software and submit that software for grading.
3. Describe algorithms to perform "simple" tasks such as numeric computation, searching and sorting, choosing among several options, string manipulation, and use of pseudo-random numbers in simulation of such tasks as rolling dice.
4. Write readable, efficient and correct C/C++ programs that include programming structures such as assignment statements, selection statements, loops, arrays, pointers, console and file I/O, structures, command line arguments, both standard library and user-defined functions, and multiple header (.h) and code (.c or .cpp) files.
5. Use commonly accepted practices and tools to find and fix runtime and logical errors in software.
6. Describe a software process model that can be used to develop significant applications composed of hundreds of functions.
7. Perform the steps necessary to edit, compile, link and execute C/C++ programs.

ABET PROGRAM OUTCOMES

Computer Engineering Students:

1. An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.

2. An ability to acquire and apply new knowledge as needed, using appropriate learning strategies

Computer Science Students:

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
3. Apply computer science theory and software development fundamentals to produce computing-based solutions.

Information Technology Students:

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

ADA STATEMENT

The University of North Texas makes reasonable academic accommodation for students with disabilities. Students seeking reasonable accommodation must first register with the Office of Disability Accommodation (ODA) to verify their eligibility. If a disability is verified, the ODA will provide you with a reasonable accommodation letter to be delivered to faculty to begin a private discussion regarding your specific needs in a course. You may request reasonable accommodations at any time, however, ODA notices of reasonable accommodation should be provided as early as possible in the semester to avoid any delay in implementation. Note that students must obtain a new letter of reasonable accommodation for every semester and must meet with each faculty member prior to implementation in each class. Students are strongly encouraged to deliver letters of reasonable accommodation during faculty office hours or by appointment. Faculty members have the authority to ask students to discuss such letters during their designated office hours to protect the privacy of the student. For additional information see the Office of Disability Accommodation website at http://www.unt.edu/oda. You may also contact them by phone at 940.565.4323.

ACCEPTABLE STUDENT BEHAVIOR

Student behavior that interferes with an instructor's ability to conduct a class or other students' opportunity to learn is unacceptable and disruptive and will not be tolerated in any instructional forum at UNT. Students engaging in unacceptable behavior will be directed to leave the classroom and the instructor may refer the student to the Dean of Students to consider whether the student's conduct violated the Code of Student Conduct. The university's expectations for student conduct apply to all instructional forums, including university and electronic classroom, labs, discussion groups, field trips, etc. The Code of Student Conduct can be found at http://deanofstudents.unt.edu.

GRADING POLICY

Your course grade will be a weighted average according to the following:

- Participation Activities 5.0%
- Challenge Activities 10.0%
- Lab Assignments 10.0%
- Projects 1 – 4 (2.5% each) 10.0%

- Lab Assessments 1 – 4 (12.5% each) 50.0%
- Comprehensive Final Exam 15.0%

Note: You must pass BOTH the exam portion (assessments and final exam) AND the non-exam portion (participation activities, challenge activities, lab assignments, and projects) with a grade of D or better in order to pass this course. Hence, an overall average greater than 60% may still result in a failing grade in some cases.

Grades will be posted on Canvas throughout the semester to provide an ongoing assessment of student progress, though final assessment will be measured using the weighted average above. Once a grade is posted on Canvas, students have two (2) weeks to dispute the grade. The proper channel for grade disputes is to first go to the original grader (either the TA or IA) in an attempt to resolve the issue. If, however, a resolution cannot be reached between the student and the grader, the student shall then go to the instructor who will have the final say on the grade.

Participation and Challenge Activities: The participation and challenge activities grades will be based on the timely completion of assigned Participation Activities and Challenge Activities, respectively, in the required zyBook e-book.

Lab Assignments: Lab assignments will be assigned and completed during the assigned lab section. Unless otherwise instructed by the instructor or Lab TA, students must be present in the lab classroom and complete each component of the lab during the scheduled lab time to receive credit for the lab. Each lab assignment will be graded using the average of all lab components based on a 0/50/75/100 scale for each component. In lab components where comments are required, but are either missing or unsatisfactory will result in the next lower grade being assigned (i.e., a 75 will become a 50, etc.) for that lab component. A missed lab due to tardiness or absence may result in a grade of 0 for the missed lab. There will be no make-up labs, though the lowest lab assignment grade will be dropped.

Projects: There will be approximately four programming assignments assigned during the semester. These programming assignments will be accepted up to 24 hours late and be assessed a 50% grade reduction penalty. Programming assignments submitted more than 24 hours late will not be accepted and receive a grade of 0.

Assessment Exams: There will be four assessment examinations given in this course. These exams will be given during the assigned lab section to assess the student's programming ability. The dates of these exams will be posted on Canvas and/or announced in class at least one week prior to the date of the exams. A make-up exam will be given at the discretion of the instructor when a student misses an exam with an excused absence. Unexcused absences on the date of an exam may result in a grade of 0 for the missed exam, so every effort should be made to attend class on the day of a scheduled exam.

Final Exam: There will be a comprehensive final exam during the scheduled exam time on Wednesday, December 12, 2018, from 1:30 PM to 3:30 PM. All students are expected to take the final exam during the scheduled time period.

ATTENDANCE POLICY

Lecture Section: Class attendance is regarded as an obligation as well as a privilege. All students are therefore expected to attend each class meeting. A student who misses class is still responsible to find out what was discussed and to learn the material that was covered and obtain the homework that was assigned on the missed day. The instructor is not responsible for re-teaching material missed by a student who did not attend class. Therefore, each student is accountable for and will be evaluated on all material covered in this course, regardless of attendance. If there are extenuating circumstances preventing you from attending the class, please notify your instructor so that you can work together to ensure your success in learning the material.

Lab Section: Students are expected to attend and be on time for their assigned weekly lab section. Missing or being tardy to a lab may result in a zero or a lower-than-usual grade for a lab. If you anticipate being unable to attend your regular lab section with a valid excuse, you must contact your instructor in advance of your lab section and before the lab is closed so that an alternate lab section may be scheduled. Failure to do so may result in a zero for the lab, but please keep in mind that the lowest lab grade will be dropped. The instructor has the final say as to whether or not an absence is excused.

ACADEMIC INTEGRITY

This course follows UNT's policy for Student Academic Integrity that can be found at https://policy.unt.edu/policy/06-003 as well as the Cheating Policy for the Department of Computer Science and Engineering (posted on Canvas). Specifically, the first instance of a student found to have violated the academic integrity (i.e., cheating) policy will result in a grade of "F" for the course and have a report filed into the Academic Integrity Database, which may include additional sanctions. Collaboration with other students is only acceptable for lab assignments that are not given as part of an exam. And although you may seek assistance from your TA/IA, Peer Mentors, and other students during the lab session for non-exam lab assignments, you are still required to work on your own lab assignment and turn in your individual work to Canvas before the lab session is complete, unless directed otherwise. Individual programming assignments (i.e., projects) given outside of the lab in this course are meant to be problem-solving exercises and must be the sole work of the individual student. You should not work with other students on shared program solutions or use program solutions found on the Internet. Specifically, you should never copy someone else's solution or code, and never let a classmate examine your code. A sophisticated program will be used to compare your work to the work of all other students (including students in past classes). If you are having trouble with an assignment, please consult with your instructor, TAs, IAs, or Peer Mentors associated with the class. You must do your own work on participation and challenge assignments as well as exams. There should be no ambiguity here. In case the above description and in-class discussion of appropriate and inappropriate collaboration do not answer all of your questions, please meet with your instructor and look at the university Student Rights and Responsibilities web page.

STUDENT RESPONSIBILITY

Students are responsible for submitting the correct assignments (i.e., uploading the proper files) for each applicable assignment submission on Canvas. In certain cases, when an assignment is submitted on time, but to an incorrect assignment location (e.g., submitting Lab 04 to Lab 05

location on Canvas), the assignment may be assessed a 30% reduction penalty if the due date has passed. If you have any questions or concerns about your submission, please work with your instructor, TA, IA, or Peer Mentor to ensure the correct file(s) is/are submitted.

SYLLABUS REVISIONS

This syllabus may be modified as the course progresses should the instructor deem it necessary. Notice of changes to the syllabus shall be made through Canvas and/or class announcement.