CSCE 1040 Computer Science II  
Syllabus, Summer 2023 10W

<table>
<thead>
<tr>
<th>Instructor:</th>
<th>Dr. Amar M. Maharjan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office:</td>
<td>NTDP E260J</td>
</tr>
<tr>
<td>Phone:</td>
<td>940-703-3172</td>
</tr>
<tr>
<td>Email:</td>
<td><a href="mailto:amar.maharjan@unt.edu">amar.maharjan@unt.edu</a></td>
</tr>
<tr>
<td></td>
<td>• Include CSCE 1040.001 in the subject line.</td>
</tr>
<tr>
<td></td>
<td>• Always use your official UNT email address.</td>
</tr>
<tr>
<td>Locations:</td>
<td>MoWe 9:30AM - 11:20AM Section 001 NTDP B142</td>
</tr>
<tr>
<td>Office Hours:</td>
<td>MoWe 11:30AM - 12:30PM in NTDP E260J and by appointment</td>
</tr>
<tr>
<td>Zoom link:</td>
<td><a href="https://unt.zoom.us/j/7033013440">https://unt.zoom.us/j/7033013440</a></td>
</tr>
</tbody>
</table>

TAs/IAs: See canvas page “TAs/IAs: Office and Labs Information”

Course Catalog Description  
CSCE 1040, the second course in the introductory sequence, focuses on more advanced C programming, designing, and implementing larger software projects, introduction to dynamic data structures, and a beginning exploration of Object-Oriented paradigms using C++. The main focus is on developing students’ software development skills.

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. Write readable, efficient, and correct C++ programs for all programming constructs defined for Programming Fundamentals I plus dynamic memory allocation, bit manipulation operators, exceptions, classes, and inheritance.
2. Design and implement recursive algorithms in C/C++.
3. Use common data structures and techniques such as stacks, queues, linked lists, trees, and hashing.
5. Use a symbolic debugger to find and fix runtime and logical errors in C software.
6. Using a software process model, design and implement a significant software application in C++. Significant software in this context means a software application with at least five files, ten functions and a make file.
7. Implement, compile, and run C++ programs that include classes, inheritance, virtual functions, function overloading and overriding, as well as other aspects of Polymorphism.
**Expected Student Outcomes:**
Student Outcomes are measurable achievements to be accomplished by the completion of the degree. These outcomes are evaluated as part of our ABET accreditation process.

**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 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 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.

**Textbook:**
We are a zyBooks text this semester as the required textbook.

zyBooks code: **UNTCSCE1040MaharjanSummer2023**
1. Click any zyBooks assignment link (Connect to zyBooks) in your learning management system (Do not go to the zyBooks website and create a new account)
2. Subscribe

You must subscribe using your UNT e-mail so that it can be tracked back to Canvas as you must access all assignment links to zyBooks through Canvas. Any issues with your zyBooks account (such as not recording grades on Canvas for completed zyBooks assignments) can be addressed by sending an email to support@zybooks.com and detailing the issue.

**Recommended Reference Book**
Walter Savitch, Problem Solving with C++, 9th Edition, Addison-Wesley

**ISBN-10:** 0133591743

**ISBN-13:** 978-0133591743
Course Evaluation
The Student Perception of Teaching (SPOT) survey is a requirement for all organized undergraduate classes at UNT. This short survey will be made available to you at the end of the semester, providing you a chance to comment on how this class is taught. I am very interested in the feedback I get from students, as I work to continually improve my teaching. I consider SPOT to be an important part of your participation in this class.

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 Access (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, refer to the Office of Disability Access website (https://www.unt.edu/oda). You may also contact ODA 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 https://studentaffairs.unt.edu/dean-of-students/conduct

Emergency Notification & Procedures:
UNT uses a system called Eagle Alert to quickly notify students with critical information in the event of an emergency (i.e., severe weather, campus closing, and health and public safety emergencies like chemical spills, fires, or violence). In the event of a university closure, please refer to Canvas for contingency plans for covering course materials or change in calendar per the Emergency Notifications and Procedures Policy.
Grading Policy
The various components of your grade are weighted as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab Assignments</td>
<td>15%</td>
</tr>
<tr>
<td>Participation Assignments (zyBooks, and in class quizzes/activities)</td>
<td>10% drop 1</td>
</tr>
<tr>
<td>Challenge Assignments (zyBooks)</td>
<td>10%</td>
</tr>
<tr>
<td>Programming Assignments Homework</td>
<td>20%</td>
</tr>
<tr>
<td>Assessment Tests in Lab (3)</td>
<td>35%</td>
</tr>
<tr>
<td>Comprehensive Final Exam</td>
<td>10%</td>
</tr>
</tbody>
</table>

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 (but not in the last assignment/homework or final exam usually), unless otherwise instructed. 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.

**Lab Assignments**: Students are responsible for submitting the correct assignments (i.e., uploading the proper files) for each applicable assignment submission on Canvas or zyBooks. A missed lab due to tardiness or absence in the lab may result in a grade of 0 for the missed lab. The lowest lab assignment grade will be dropped.

**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 zyBooks e-book. **Pay special attention to see that your zyBooks scores match your Canvas scores for these activities.** Your assignments are autograded, so your Canvas score must populate immediately after you submit on zyBooks. If they don’t, start the activity on Canvas and submit again.

**Homeworks**: Programming assignments will not be accepted late beyond a 30-minute grace window unless there are verifiable system problems or outages of service on Canvas. Partial credit will only be given for programs which compile but which are not complete. **No regular or late credit will be given for programs that do not compile on the CSE Linux servers or in the E-Book environment as assigned!** The only points you may earn for non-compiling assignments are any points for design or report components. If you are having trouble with an assignment, please consult with your instructor, TAs, IAs, or Peer Mentors associated with the class.

**Assessment Exams**: There will be three assessment examinations given on 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 (online on canvas) during the scheduled exam time: **001 (Friday July 28, 2023).**
All students are expected to take the final exam during the scheduled time.

Course Policies

**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 for finding 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.

If you are experiencing any symptoms of COVID-19 ([https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html](https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html)) please seek medical attention from the Student Health and Wellness Center (940-565-2333 or askSHWC@unt.edu) or your health care provider PRIOR to coming to campus. As of April 29, 2022, the university will no longer offer contact tracing for positive test cases. Individuals who test positive for the virus should follow current CDC guidelines for quarantine/isolation and follow the treatment advice of their primary care physician. See [https://studentaffairs.unt.edu/student-health-and-wellness-center/services/covid](https://studentaffairs.unt.edu/student-health-and-wellness-center/services/covid) for more.

**Lab Section:** Students are required to attend their assigned weekly lab section. 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.

**Make-up Labs:** You are not permitted to go to a different lab section than the one you are registered for. However, if you have to miss your lab section due to unavoidable circumstances such as health or other university engagements, you may be granted permission to complete your lab in a different lab section with instructor permission. To request permission, you need to email the instructor with the following information:

- **Which lab are you missing?** Provide date, time and lab section number.
- **Why are you missing the lab?** Attach documentation such as receipts, doctor’s note, etc. for proof.
- **When do you want to make up the lab?** Provide date, time and lab section number. A list of available labs to choose from is available in the Lab Schedule spreadsheet on Canvas. Make sure you choose a lab with small enrollment so that you can be accommodated in the lab.

Requests for make-up labs without this information will not be considered.
**Programming Assignments:** Starting early on programming projects is strongly encouraged. Students typically have great difficulty completing their projects in one night the day before they are due. Students are allowed to discuss program design and other high-level issues with each other. Students are also allowed to help each other understand specific compiler or runtime error messages. Copying all or part of another person's program is strictly prohibited and will be treated as cheating. All programs will be submitted through the class Canvas website. No assignments will be accepted for grading via email or on paper.

**Communications:** The instructor, peer mentors and TAs require a current copy of the program when a student is asking a question about a program. Questions must be specific. Please do not send a program and simply say “This does not work”. Also please be prepared to show and explain your design when asking questions. If you have not completed a design then the TA, Peer Mentor or Instructor may require you to complete one before answering questions about program logic for your code.

**Course Status:** All pertinent information about the class (assignments, exam reviews) is available via the class webpage. If there is ever a question as to when something is due, or an additional copy of a course document is needed, ALWAYS check the class webpage. If you feel there is incorrect or missing information on the class website, email the instructor about the problem immediately. Electronic mail (email) will be a major means of communication with the instructor outside of actual classroom discussions. Please keep this information sheet handy during the semester and always periodically check the class homepage for any course information, including scheduling of programming assignments, exams, and exam reviews.

**Class Notes and Programming Examples**
The lecture slides will be available on Canvas. The programming examples provided in the class will be on the CSE servers inside my public folder. You will need an SSH client to access the server and an SFTP client to download my examples to your personal computer. An announcement will be posted on Canvas for future reference. There is also a document named **Accessing CSE Servers** on Canvas that shows you how to access these programming examples. You can also bring your own computer to the class and type along with me during the programming demonstrations.

**Academic Integrity**
This course follows UNT’s policy for Student Academic Integrity that can be found at [https://policy.unt.edu/policy/06-003](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. 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 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.

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.

Tentative Class Schedule (subject to change):

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Material Covered</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>05/22 - 05/26</td>
<td>Syllabus, M1: Introduction and Review (Structures, Pointers, Dynamic Memory)</td>
<td>Lab0, Lab1</td>
</tr>
<tr>
<td>2</td>
<td>05/29 - 06/02</td>
<td>M1: Introduction and Review (Structures, Pointers, Dynamic Memory), M2: Classes and Objects</td>
<td>Lab1, Lab2</td>
</tr>
<tr>
<td>3</td>
<td>06/05 - 06/09</td>
<td>M2: Classes and Objects, M3: Constructors and Destructors, Copying objects, Review</td>
<td>Lab3, Lab4</td>
</tr>
<tr>
<td>4</td>
<td>06/12 - 06/16</td>
<td>M4: Standard Template Library</td>
<td>Lab5, Lab6 (Exam)</td>
</tr>
<tr>
<td>5</td>
<td>06/19 - 06/23</td>
<td>M4: Standard Template Library, M5: Data Structures and Tools, Review</td>
<td>Lab7, Lab8</td>
</tr>
<tr>
<td>6</td>
<td>06/26 - 06/30</td>
<td>M5: Data Structures and Tools, M6: Recursion</td>
<td>Lab9, Lab10 (Exam)</td>
</tr>
<tr>
<td>7</td>
<td>07/03 - 07/07</td>
<td>M7: Inheritance</td>
<td>Lab11, Lab12</td>
</tr>
<tr>
<td>8</td>
<td>07/10 - 07/14</td>
<td>M7: Inheritance, M8: Exceptions, Review</td>
<td>Lab13, Lab14</td>
</tr>
<tr>
<td>9</td>
<td>07/17 - 07/21</td>
<td>M8: Exceptions, M9: Class and Function Templates</td>
<td>Lab15 (Exam)</td>
</tr>
<tr>
<td>10</td>
<td>07/24 - 07/28</td>
<td>Final Exam Review</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>

Important Dates:
05/22: First day of term
05/29: Memorial Day – No Classes
06/19: Juneteenth Observance - No Classes
07/04: Independence Day- No Classes
07/28: Last day of term