Instructor: Dr. Mark A. Thompson
Office: NTDP F264
Telephone: (940) 369-7055
E-mail Address: Mark.Thompson2@unt.edu
Class Location/Time: NTDP B140, MoWe 8:30 – 10:50 AM
Office Hours: MoWe 1:30 – 2:30 PM or by appointment

Every attempt will be made to answer e-mails within 24 hours. Please include CSCE 1040.021 (or your specific lab section) in subject line.


Blackboard
This course will use Blackboard, a Web-based course management system, to distribute course materials, communicate and collaborate online, post grades and submit assignments. You are responsible for checking the Blackboard course site regularly for class work and announcements.

COURSE 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.
TOPICS (subject to change)
1. Review
2. Dynamic Memory Management in C and C++
3. Recursion
4. Stacks and Queues
5. Linked Lists
6. Hashing
7. Basic Trees
8. Bit Manipulation
9. Make files
10. Debugging and Exceptions
11. Storage Classes
12. Object Oriented Design
13. Classes and Objects
14. Polymorphism
15. Inheritance
16. STL
17. Friends
18. Virtual Functions
19. Copy constructors
20. Function Overloading
21. Operator Overloading

ADA STATEMENT
The University of North Texas makes reasonable academic accommodation for students with disabilities. Students seeking 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 an accommodation letter to be delivered to faculty to begin a private discussion regarding your specific needs in a course. You may request accommodations at any time, however, ODA notices of 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 accommodation for every semester and must meet with each faculty member prior to implementation in each class. For additional information, see the Office of Disability Accommodation website at http://disability.unt.edu. 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.
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.

Excessive student absences may have a negative impact on a student’s comprehension and learning. Since this class meets on a MoWe schedule, students with more than two (2) absences may have their participation/attendance grade lowered for each absence beyond the initial two absences. Therefore, I expect your participation and attendance in this class to receive high priority. Note that unless there are extenuating circumstances requiring an absence lasting more than two (2) class periods, all absences, whether excused or not, will be counted.

Lab Section: Students are expected to attend and be on time for their weekly lab section. Missing or being tardy to a lab may result in a zero or a lower-than-usual grade for a lab and/or quiz. 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 and/or quiz, but please keep in mind that the lowest lab and lowest quiz grade will be dropped. The instructor has the final say as to whether or not an absence is excused.

ACADEMIC DISHONESTY

This course follows the Department of Computer Science and Engineering Cheating Policy. Specifically, students caught cheating or plagiarizing will receive a “0” for that particular assignment or exam for the first offense. Additionally, the incident may be reported to the Dean of Students, who may impose a further penalty. A second instance of cheating in this class will result in a grade of F in the class, and referral to the Department Chairperson and Dean of Engineering, whereby a dismissal hearing may be initiated by the Dean of Engineering.

Collaboration with other students is only acceptable for lab programs that are not given as part of an exam. And although you may seek assistance from your TA, IA, and other students during the lab session for non-exam lab programs, you are still required to work on your own lab program and turn in your individual work to Blackboard before the lab session is complete or as directed otherwise.

Individual programming assignments (i.e., homework) 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, or IAs.

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.
Grading Policy

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

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance/Participation</td>
<td>5.0%</td>
</tr>
<tr>
<td>Lab Programs</td>
<td>25.0%</td>
</tr>
<tr>
<td>Quizzes</td>
<td>10.0%</td>
</tr>
<tr>
<td>Programming Assignments</td>
<td>25.0%</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>15.0%</td>
</tr>
<tr>
<td>Comprehensive Final Exam</td>
<td>20.0%</td>
</tr>
<tr>
<td>Total</td>
<td>100.0%</td>
</tr>
</tbody>
</table>

Grades will be posted on Blackboard 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 assigned on Blackboard, students have one (1) week 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.

Attendance/Participation: Attendance/Participation grades will be based on attendance and contribution to in-class discussions. Disruptive behavior and absences deemed excessive will result in a lower attendance/participation grade. Additionally, absences beyond what is calculated for the attendance/participation grade may result in a grade of WF for nonattendance or a one letter grade reduction in the final grade assigned.

Lab Programs: Lab programs will be assigned with all or most components completed during the assigned lab section. 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 will be graded using the average of all lab components based on a 0/50/100 scale for each component. In lab components where comments are required, but are either missing or unsatisfactory, the entire lab program grade will incur a 10% grade reduction penalty. There will be no make-up lab programs. However, the lowest lab program grade will be dropped.

Quizzes: Quizzes will be given at the beginning of each laboratory session. A missed quiz due to tardiness or absence may result in a grade of 0 for the missed quiz. There will be no make-up quizzes. However, the lowest quiz grade will be dropped.

Programming Assignments: There will be approximately four non-lab 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.

Midterm Exam: There will be a midterm examination given in this course that will consist of a coding portion and a “paper” portion that will be given during the lecture period. The date of this exam will be posted on Blackboard and announced in class at least one week prior to the date of the exam. 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 on Wednesday, July 6, 2016, during our regularly scheduled lecture time. As with the midterm, a coding portion will be given during the lecture section along with the “paper” portion. All students are expected to take the final exam during the scheduled time period.
STUDENT RESPONSIBILITY

Students are responsible for submitting the correct assignments (i.e., uploading the proper files) for each applicable assignment submission on Blackboard. When an incorrect assignment is submitted (or no assignment is submitted) to Blackboard, students wanting to submit with the correct file(s) will have their assignment assessed a 30% reduction penalty. If you have any questions or concerns about your submission, please work with your instructor, TA, or IA 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 Blackboard 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>5/16 – 5/20</td>
<td>Intro, Review</td>
<td>L1, P1</td>
</tr>
<tr>
<td>2</td>
<td>5/23 – 5/27</td>
<td>Sec 4.6, Ch 10, Ch 14, Bit/Byte Manipulation</td>
<td>L2, Q1</td>
</tr>
<tr>
<td>3</td>
<td>5/30 – 6/3</td>
<td>Ch 11, Dynamic Memory Management</td>
<td>L3, Q2, P2</td>
</tr>
<tr>
<td>4</td>
<td>6/6 – 6/10</td>
<td>Ch 12, Makefiles, Ch 16, Review</td>
<td>L4, Q3</td>
</tr>
<tr>
<td>5</td>
<td>6/13 – 6/17</td>
<td>Hashing, Ch 13</td>
<td>Midterm, L5, Q4, P3</td>
</tr>
<tr>
<td>6</td>
<td>6/20 – 6/24</td>
<td>Ch 13, Ch 15</td>
<td>L6, Q5</td>
</tr>
<tr>
<td>7</td>
<td>6/27 – 7/1</td>
<td>Ch 18</td>
<td>L7, Q6, P4</td>
</tr>
<tr>
<td>8</td>
<td>7/4 – 7/8</td>
<td>Review</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>

IMPORTANT DATES

May 16          First Class Day
May 30          Memorial Day (university closed)
Jun 7           Last day to drop a course with a grade of W for courses a student is not passing
Jun 16          Last day to drop a course with written consent of instructor
Jul 4           Independence Day (university closed)
Jul 6           Last Class Day and Final Exam