INSTRUCTOR: Dr. Mark A. Thompson
Office: NTDP F264
Telephone: (940) 369-7055
E-mail Address: Mark.Thompson2@unt.edu
Class Location/Time: NTDP B185, MoWeFr 8:30 – 9:20 AM
Office Hours: MoWe 9:30 – 10:30 AM
TuTh 1:00 – 2:00 PM
or by appointment

Every attempt will be made to answer e-mails within 24 hours. Please include CSCE 3600.00x (your specific 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 3600 maintains a focus on systems programming, both from the standpoint of learning about how computer system software works and learning/improving students’ programming skills in K&R C and bash. The course will include one or more major programming projects that will be completed in groups as well as several smaller programming assignments to be done individually.

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 robust, efficient, readable and correct system software using the C programming language.
2. Demonstrate an understanding of processes and threads by developing applications using multiple processes and multi-threaded activities in a Linux environment.
3. Demonstrate an understanding of deadlocks and synchronization through the development of application(s) that utilize a variety of mutual exclusion mechanisms.
4. Develop shell scripts and utilities that demonstrate an understanding of memory, file and process management and interaction, including concepts such as virtual memory and disk scheduling.
5. Create a Linux-based application that utilizes inter-process communication mechanisms such as pipes and sockets to communicate information between independently running processes on one or multiple platforms.
6. Demonstrate an understanding of the use and interaction among compilers, macro processors, assemblers, linkers and loaders through their use in creating the applications described in previous outcomes.
TOPICS (subject to change)
1. Unix/Linux shell
2. Linux programming utilities
3. Shell programming
4. Files
5. Processes
6. Interaction of compilers, macro processors, assemblers, linkers, loaders, and operating systems to run a high-level program
7. Compilers
8. Operating system functions
9. Inter-process communication
10. Concurrent programming
11. Memory management

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

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 MoWeFr schedule, students with more than five (5) absences may have their participation/attendance grade lowered for each absence beyond the initial five 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.

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.

This course will contain both group assignments as well as individual assignments, so you should be absolutely aware of the assignment requirements before starting an assignment. On major programming assignments, you are to work in a group as directed by your instructor. On minor programming assignments, you are to work alone (i.e., individually) unless explicitly directed otherwise by your instructor. Do NOT work with other students on shared program solutions. Do NOT get help with algorithms or coding from anyone other than your instructor, TA, IA (i.e., Grader), or Peer Mentor assigned to this course. Do NOT use partial program solutions from the Internet unless those partial solutions are provided as part of the assignment description. This means that individual programming assignments must be the sole work of the individual student. If you are having trouble with an assignment, please consult with your instructor, TAs, IAs, or peer mentors. Failure to adhere to these strict standards may be cause for disciplinary action even leading to expulsion from the University.

It IS permissible, however, to obtain help from whomever you wish to fix syntax errors, but please remember that for anything except syntax errors, getting programming assistance from any source other than mentioned above will be considered academic misconduct and be treated accordingly.

In the case that the above description or any 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>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance/Participation</td>
<td>6.0%</td>
</tr>
<tr>
<td>Minor Programming Assignments</td>
<td>24.0%</td>
</tr>
<tr>
<td>Major Programming Assignments</td>
<td>25.0%</td>
</tr>
<tr>
<td>Midterm Exams 1 – 2</td>
<td>25.0% (12.5% each)</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20.0%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100.0%</strong></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.

Most programming assignments will be due at 11:59 PM on the specified due date to Blackboard. No late assignments will be accepted. All assignments must be completed and submitted according to their specific directives.

**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.

**Minor Programming Assignments:** Minor programming assignments will be assigned based on the material from the lectures and textbook. These are meant to be individual programming assignments, so you should work on these alone unless explicitly directed otherwise by your instructor.

**Major Programming Assignments:** Major programming assignments will be worked on in a group and are meant to be more comprehensive problem solving exercises based on the material from the lectures and textbook.

**Midterm Exams:** There will be two midterm examinations given in this course. The dates of these exams will be posted on Blackboard and 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 on Monday, May 9, 2016, from 8:00 AM to 10:00 AM. 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 to Blackboard, students wanting to resubmit 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, IA, or Peer Mentor as appropriate 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>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/18 — 1/22</td>
<td>Intro, Parsing</td>
</tr>
<tr>
<td>2</td>
<td>1/25 — 1/29</td>
<td>Bash</td>
</tr>
<tr>
<td>3</td>
<td>2/1 — 2/5</td>
<td>Processes</td>
</tr>
<tr>
<td>4</td>
<td>2/8 — 2/12</td>
<td>Concurrency</td>
</tr>
<tr>
<td>5</td>
<td>2/15 — 2/19</td>
<td>Review, Exam 1</td>
</tr>
<tr>
<td>6</td>
<td>2/22 — 2/26</td>
<td>Streams, Pipes, Files</td>
</tr>
<tr>
<td>7</td>
<td>2/29 — 3/4</td>
<td>Sockets</td>
</tr>
<tr>
<td>8</td>
<td>3/7 — 3/11</td>
<td>Program Execution</td>
</tr>
<tr>
<td>9</td>
<td>3/14 — 3/18</td>
<td>Spring Break</td>
</tr>
<tr>
<td>10</td>
<td>3/21 — 3/25</td>
<td>Compilers</td>
</tr>
<tr>
<td>11</td>
<td>3/28 — 4/1</td>
<td>Review, Exam 2</td>
</tr>
<tr>
<td>12</td>
<td>4/4 — 4/8</td>
<td>Mutual Exclusion</td>
</tr>
<tr>
<td>13</td>
<td>4/11 — 4/15</td>
<td>Virtual Memory</td>
</tr>
<tr>
<td>14</td>
<td>4/18 — 4/22</td>
<td>Dynamic Memory Management</td>
</tr>
<tr>
<td>15</td>
<td>4/25 — 4/29</td>
<td>TBD</td>
</tr>
<tr>
<td>16</td>
<td>5/2 — 5/6</td>
<td>Review</td>
</tr>
<tr>
<td>17</td>
<td>5/9 Mon</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>

IMPORTANT DATES
Jan 19         First Class Day
Feb 26         Last day to drop a course with a grade of W for courses a student is not passing.
Mar 14 - 20    Spring Break (no classes)
Apr 5          Last day to drop a course with written consent of instructor.
May 5          Last Class Day
May 9          Final Exam