CSCE 2110 Syllabus

Spring 2017

Instructor: Fahmida Hamid
Office: NTDP F214
Phone: 940-369-7223
Office Hours: Monday (10:00 AM – 11:00 AM)
Tuesday (1:00 PM – 2:00 PM)
Email: Fahmida.Hamid@unt.edu

Peer Mentor(s):
Name: Scarlett Jones
Email: scarlettjones@my.unt.edu
Office Hours:
Monday 9:30 AM - 11:30 AM & 12:30 PM - 1:30 PM
Wednesday 10:30 AM - 12:30 PM & 1:30 PM - 2:30 PM
Thursday 5:00 PM - 8:00 PM (location for evening hours subject to change)
Friday 9:30 AM - 10:30 AM
(Note: Hours to be held in the Discovery Park Library, unless it is the evening hours on Thursday, because the library closes at 6:30 PM)

Name: Jonathan Roosa
Email: JonathanRoosa@my.unt.edu
Office:
NTDP B135 (inside of the DP library), Thursday evening hours office TBD
Office Hours:
Tuesday and Thursday from 10:00 AM to 1:30 PM,
and Thursday evening from 5:00 PM to 8:00 PM
(location for evening hours is subject to change)

Teaching Assistants:
Name: Venkata Prasanth Yanambaka
Email: vy0017@unt.edu

Name: Bachu Saikrishna
Email: saikrishnabachu@my.unt.edu
Recitation:
CSCE2110.205, NTDP B158, Wed 12:30 pm - 1:20 pm
Name: Tongtian Ma  
Email: tongtianma@my.unt.edu  
Recitation:  
CSCE 2110.203 Friday 10:30-11:20 at B157  
CSCE 2110.204 Monday 11:30-12:20 at B192

Name: Niharika Pocharam  
Email: niharikapocharam@my.unt.edu  
Recitation:  
CSCE 2110.201, NTDP B157, Monday 8:30AM – 9:20AM  
CSCE 2110.202, NTDP B192, Wednesday 9:30AM – 10:20AM

Textbook: (Available on line – for free !!!)

*Foundation of Computer Science*, Aho and Ullman

Course Description:

Computing Foundations II. Introduces students to both data structures and formalisms used in computer science including 1) Sets, 2) Relational Tables, 3) Graphs, 4) Regular Expression and Automata, 5) Propositional Logic and 6) Advanced topics not included in the text.

Course Outcomes:

Course outcomes are measurable achievements to be accomplished [by the student] by the completion of a course. Outcomes are evaluated in ABET’s accreditation process.

1. A solid foundation in conceptual and formal models  
2. The ability to use abstraction in the design and implementation of algorithms  
3. Use of C++ to implement trees and lists  
4. Application of big-Oh notation to evaluate and compare algorithms  
5. Use sets, relational tables, graphs, regular and expressions in the design of software and  
6. Introduction to Computability theory and amortized analysis
Policies:

- Three exams: 2 midterms and a final, to be held in the university testing center OR using scantron forms in class. All exam questions will be multiple choice.
- All of your programs (three of them) will be written in C++. In addition, you should use a C++ that is compatible with that compiled on the “CSE” machines within the CSE department. If your program does not compile with the CSE g++ and without any special flags, it will be counted as a program that has compile errors (which it does).
  
  **EXCEPTION:** You can, if you choose, submit a makefile with your program, in which case your program will be “compiled” using the makefile. It still needs to be written in C++ but it can use whatever compile flags you like, as long as it works on the CSE machines. But be warned. If your makefile doesn’t work properly with whatever code you submit, your program will be considered to have compiler error(s).
- You will **NOT** use the Standard Template Library in your programs this term in 2110, with the exception of the **String** class. Failure to follow this “guideline” will result in a 0 for the program.
- Most programs are due at 11:59pm on the due date.
- No late programs will be graded
- All programs will be submitted to Blackboard.
- Attendance will not be taken in lecture. However, your attendance is strongly recommended to improve your opportunity to meet course outcomes. In addition, part of your grade (see grade break down below) will be based upon work done in class. Of course, if you are not in class when this work is done your grade will suffer.
- We will be using the REEF response system now supported by a UNT-wide site license. So, you should bring an “internet device” (laptop, table, smart phone, dumb phone, …) to class with you.

Grading: The components of your grade will be weighted as follows:

- Recitation Quizzes and “problems”, 10%
- In-class Exercises, 5%
- Midterm exams – two at 15% each
- 1 Final Exam, 25%
- Three C++ programs, 30%

**NOTE:** In order to pass this class (C or better) each student will have to “pass” both the program portion of the course (60% or better) AND have an overall average of at least 50%. Often students assume that I grade on a 90/80/70/60 scale. That is NOT the case. As far as the “overall” score is concerned I will guarantee that if you meeting the 60% minimum on program scores AND your overall average is 90% or higher you’ll receive an ‘A’. If it is 80% or higher you’ll receive at least a “B”. If it is 70% or higher you’ll receive at least a “C”.

**Beyond that I give no guarantees, except that if your programming score is less than 60, you’ll fail not matter your overall average.**
Make-Up Policy:

There will be no make-up exams, recitations, or programs given in this class. However, for documented excused absences or emergencies on a day of an exam or a recitation, the exam and/or recitation grades will be replaced by an average of the other exams or recitation scores. There is one exception to this rule. Under NO circumstances will more than one exam or recitation score be replaced by an average of the other scores. For a second (and subsequent) missed exam or lab, even if all are excused, students will receive a 0 for the missed work.

Excused Absences

Students are expected to schedule routine appointments and activities to not conflict with attending class. However, some absences cannot be prevented. In the event of a medical emergency or family death, students must request an excused absence as quickly as feasible following the emergency. Use common sense. Students must provide documentation to verify the emergency.

Emergencies

By definition, emergencies cannot be planned for. Your instructor attempts to make accommodations in these instances that allow for making up missed work and completion of the course in a timely manner. Among these emergencies are

- A death in your immediate family
- An accident or illness requiring immediate medical treatment and where a doctor has indicated attending class is impossible or inadvisable.
- Employees on call 24/7 must document that they were called during class time.

Collaboration and Cheating:

Programming:

Do NOT work with other students on shared program solutions. Do NOT get help with algorithms or coding from anyone other than Dr. Sweany or the 2110 TA, or peer mentor. Do NOT use even partial program solutions from the internet unless those partial solutions are provided to you by Dr. Sweany as part of the assignment description. Failure to adhere to these strict standards will be cause for disciplinary action that could be as severe as expulsion from the university. Those who are deemed to have cheated on program will receive a penalty that exceeds getting a 0 score for the program. Of course students who are deemed to have cheated have an appeal process of any penalty I invoke. If and when the time comes that appeal process will be described to the student(s) in question.

It IS permissible to obtain help from whoever you wish to fix syntax errors. But remember, for anything but syntax errors, getting programming assistance from any source other than Dr. Sweany or the 2110 TAs or peer mentor(s) will be considered cheating and dealt with harshly.

NOTE: I’m adding a new caveat to my program grading policy this term, due to a “problem” that arose in Spring 2016. A student completed a programming assignment and placed the solution on
GitHub. Another student accessed the GitHub repository, copied the solution and then made it available for other students to use. Thirteen students thus turned in a copy of the same solution and were penalized. At least one is no longer a UNT student. I subsequently learned that the department has had several cases where solutions placed on GitHub found their way into submissions by students who did NOT write the code. So, in future, students in my class who place their solutions on GitHub or any other unprotected site will be held equally responsible as any student who submits a copy of that program. If you wish to debate the use of GitHub, drop by my office during office hours and we’ll have a spirited conversation about it.

**Recitations and “Lectures”**

In stark contrast to programs, quizzes and labs I STRONGLY encourage students to work in groups both in the “lecture” classes and in recitation. I’ll explain my reasoning in class but let me say here that if you don’t take advantage of the cooperative learning available class and in recitation you’ll be missing out both on learning foundations in 2100 and in any future career you might pursue.

**Exams:**

And, of course you need to do your own work on quizzes and exams as well. Here there should be no ambiguity at all.

In case the above description, and in-class discussion of my views on appropriate and inappropriate collaboration does not answer all of your questions, please look at the university Student Rights and Responsibilities web page.

**SPOT**

The Student 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 the SPOT to be an important part of your participation in this class.

**ADA**

UNT complies with all federal and state laws and regulations regarding discrimination, including the Americans with Disability Act of 1990 (ADA). If you have a disability and need a reasonable accommodation for equal access to education or services please contact the Office of Disability Accommodation.
## Tentative Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
<th>Reading</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/17/17</td>
<td>Intro, Sets</td>
<td>Chapter 7</td>
<td></td>
</tr>
<tr>
<td>1/24/17</td>
<td>Sets</td>
<td>Chapter 7</td>
<td>“Assignment 0”, 1/27/17</td>
</tr>
<tr>
<td>1/31/17</td>
<td>Sets</td>
<td>Chapter 7</td>
<td></td>
</tr>
<tr>
<td>2/7/17</td>
<td>Relational Model</td>
<td>Chapter 8</td>
<td></td>
</tr>
<tr>
<td>2/14/17</td>
<td>Relational Model</td>
<td>Chapter 8</td>
<td></td>
</tr>
<tr>
<td>2/21/17</td>
<td>Relational Model</td>
<td>Chapter 8</td>
<td>Prog1, 11:59pm 2/24/17</td>
</tr>
<tr>
<td>2/28/17</td>
<td>Exam, Graph</td>
<td>Chapter 9</td>
<td>Exam 1</td>
</tr>
<tr>
<td>3/7/17</td>
<td>Graphs</td>
<td>Chapter 9</td>
<td></td>
</tr>
<tr>
<td>3/14/17</td>
<td>Spring Break</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3/21/17</td>
<td>Graphs</td>
<td>Chapter 9</td>
<td></td>
</tr>
<tr>
<td>3/28/17</td>
<td>Graphs</td>
<td>Chapter 9</td>
<td></td>
</tr>
<tr>
<td>4/4/17</td>
<td>Regular Expressions</td>
<td>Chapter 10</td>
<td>Prog 2, 11:59pm 4/7/17</td>
</tr>
<tr>
<td>4/11/17</td>
<td>Automata</td>
<td>Chapter 11</td>
<td>Exam 2</td>
</tr>
<tr>
<td>4/18/17</td>
<td>Grammars</td>
<td>Chapter 11</td>
<td></td>
</tr>
<tr>
<td>4/25/17</td>
<td>Propositional Logic</td>
<td>Chapter 12</td>
<td></td>
</tr>
<tr>
<td>5/2/17</td>
<td>Computability</td>
<td>To be determined</td>
<td>Prog3, 11:59pm 5/4/17</td>
</tr>
<tr>
<td>5/6/13</td>
<td>Finals</td>
<td>Course Objectives</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>