ECE322C – DATA STRUCTURES – UNIQUE SECTION # 14905

COURSE SYLLABUS

Spring 2005
(based on Herb Krasner's Spring 2005 Syllabus Version 1.0)

INSTRUCTOR Sarfraz Khurshid; **EMAIL:** khurshid@ece.utexas.edu
OFFICE: ACES 5.120 **PHONE:** x1-8244
OFFICE HOURS W 2:00-4:00, and by appointment.

TA : Bolu (Moboluwaji O. Sanu) ; **EMAIL** : sanu@ece.utexas.edu
The office/laboratory hours will be posted online during the second week of classes.

CLASS MEETING SPECIFICS. MW 5-6:30, CPE 2.210

COURSE OBJECTIVES: To learn: how to program with a focus on data abstractions; how to do this in C++; the application of various common data structures and templates; and an introduction to the field of algorithm analysis. Additionally to prepare students for future subjects in ECE, software engineering or embedded/systems software.

SUBJECTS COVERED: The topics included in this course are covered roughly in this order: From C to C++, elementary data structures, C++ classes, Dynamic Arrays, Lists, Stacks, Queues, Trees, Hashing, Priority Queues, Sets, and Maps.

PREREQUISITES. EE312 with a grade of at least *C*. Incoming students are expected to know the basics of computers and computation; and how to program in C using features of the language, such as: variables and operators, builtin data types, execution control structures, pointers, arrays, screen I/O, structs , linked lists, and recursion. The student should know how to use basic programming tools and techniques, such as: a programming language development toolset and symbolic debugging. The incoming student may or may not have been introduced to subjects such as: abstract data types, analysis of algorithms, program design techniques, object-oriented programming, advanced C++ features (e.g. classes, templates, etc.).

TEXT. *Data Structures with C++ Using STL, 2nd Edition*, by W. Ford and W. Topp, Prentice Hall, 2002, ISBN 0-13-085850-1

SUGGESTED REFERENCES for those students who wish to become stronger in C++ programming: *C++ Programming Language, 3rd Edition, B. Stroustrup, Addison Wesley, 2000; Thinking in C++*, Bruce Eckel, Prentice-Hall, 2000 (2nd Edition). (Other references may be found on the EE322C homepage.)

ATTENDANCE Attendance is expected. Whether you come to class or not, you are responsible for keeping up with what happens in class. If you miss a class (other than for illness or an emergency), it is not reasonable for you to expect me to repeat just for you the material that was covered in the class that you missed. This applies both to the content of the class as well as to announcements about class policies, events, deadlines, or whatever. You can expect a loss of at least one letter grade if you miss four or more lectures.

DROPPING. I will not sign any drop card after the second week of classes unless it is approved beforehand by Dean Meyer's office. The course grade that I assign in these cases will always be my estimate of your current grade. In particular, your grade must be a 'C' or better in order to receive a 'Q' on your drop application.

COURSE GRADES. Course grades will be based on the following components.

Component	Date	Weight
Exam 1	February 23 in class (tentative)	20%
Exam 2	April 6 in class (tentative)	20%
Exam 3	May 4 in class (tentative)	20%
Assignments	(Dates are stated in each assignment) Up to 10 assignments are planned.	40%

The grade you are given, either on an individual exam or assignment or as your final grade, is not the starting point of a negotiation. It is your grade unless a concrete error has been made. Do not come to see me or the TA to ask for a better grade because you want one or you "feel you deserve it". Come only if you can document a specific error in grading or in recording your scores. Errors can certainly be made in grading, especially when many students are involved. But keep in mind that the errors can be made either in your favor or not. So it's possible that if you ask to have a piece of work re-graded your grade will go down rather than up.

Remember that the most important characteristic of any grading scheme is that it be fair to everyone in the class. Keep this in mind if you're thinking of asking, for example, for more partial credit points on a problem. The important thing is not the exact number of points that were taken off for each kind of mistake. The important thing is that that number was the same for everyone. So it can't be changed once the grading is done and the exams or assignments have been returned.

If you have questions or concerns about any of your grades, contact me during office hours or via email.

Final Grades: Final grades will be assigned according to the following standard criteria:

Final Average	Letter Grade
90 - 100	A
80 - 89	B
70 - 79	C
60 - 69	D
0 - 59	F

Final class grades will be calculated to 2 decimal places and rounded to the nearest integer. 89.49 is a B. 89.50 is an A. The line has to be drawn somewhere, and no special allowances will be made for students whose final average falls near, but below the cutoff. Nonacademic explanations for poor class performance will have no bearing on the assignment of grades.

EXAMINATIONS. Exams will cover material from lecture, assignments, and the assigned readings. Exams will be cumulative, although they will be more heavily weighted towards material not yet tested. Programming is a cumulative discipline, so it is important to master earlier topics in order to understand later topics. Exam scores may be curved if the instructor believes it is warranted. Three exams will be given in class. The dates are shown in the table above. If your work or a personal situation forces you to unexpectedly miss exams, you should expect to get a zero on those occasions. If you miss an exam because of illness, you are expected to provide a statement from a doctor stating that, in his/her opinion, it was impossible for you to attend because of illness. A slip showing you visited the UT Health Center or your personal doctor is not sufficient for this. In other situations, you should contact me beforehand.

PROGRAMMING ASSIGNMENTS. The assigned class work in this course will consist of up to ten programming assignments. Programming is a discipline that you learn by doing, not by listening to a lecturer. Therefore, doing the programming assignments is crucial to performing well in this class. Assignments will be given almost every week. Each assignment will have a clearly stated due date and time. **NO LATE ASSIGNMENTS WILL BE ACCEPTED**. If you are having considerable difficulty with the early assignments, this is a sign that you may be in over your head - you should come see me immediately. The assignments will require a substantial time commitment over several days (an average of 6 hours per week outside of class should be expected). Be sure to budget sufficient time to complete assignments before the deadline. At the time you submit each assignment for grading, you are **required** to make a backup copy of the source code file on your removable secondary storage device (e.g. a floppy or ZIP disk). This will be necessary in cases where your program gets lost, is corrupted, or if there is some dispute over what was turned in when. If you cannot finish an assignment on time, then submit whatever you have finished before the deadline to receive some partial credit.

SUBMITTING PROGRAMMING ASSIGNMENTS. Programming assignments will be submitted by the blackboard system. Your scores will also be available there.
Program assignment submission rules:

1. Upload the completed files (usually the .cpp or .h text files) that form your program.
2. Do **not** email files to the professor or to the TA.
3. When you submit your file, you will see a confirmation of your submission and you should be able to view your submission.
4. It takes about a week to grade the projects.
5. You must submit before the submission deadline. Late submissions will not be graded.
6. As part of the required documentation header block, the top four lines of each file that is submitted should be comments with the following information:

    // your name - last name first
    // your student EID
    // your email address
    // EE322C-Assignment *n* - where *n* is the assignment number (1, 2, ...)

PROGRAMMING ASSIGNMENT GRADES. Assignment grading criteria may vary on each assignment. However, in general, programs that do not compile correctly on the Lab configuration will receive no more than 25% of the possible points. Larger point deductions are given for such things as: incorrect results, missing features, bad solution logic, poor style, etc. With regard to programming style, I expect you to follow the C++ coding standards that are found on the course web page. These coding standards will be loosely enforced on the early assignments but rigorously enforced on the latter assignments. In addition the following criteria are important: (i) a block structured design should be evident; (ii) comments and/or appropriate variable names should be used to make your program readable; (iii) appropriate prompts and messages for input and output should be given to the user of your program. Unless stated otherwise, more emphasis will be placed on program clarity than on program length, speed or size. For pair or team programming assignments, each member will receive the same grade as the other(s).

PROGRAMMING LAB and SOFTWARE. The computers in the third-floor labs of ENS have all of the software needed for this course. If you plan to do all of your work at home, you will need a *C*/*C++* compiler, a web browser, an email program, and the program *Adobe Acrobat Reader*. All of the programs are free except the *C*/*C++* compiler, which can be purchased at the *Campus Computer Store* or directly from Metrowerks. Programs submitted are expected to run on the version of Metrowerks CodeWarrior that is currently installed in these labs. If your program doesn't run on the CodeWarrior version currently in the labs, you can expect to lose points even if it runs correctly on some other compiler.

GRADE DISPUTES AND CORRECTIONS. If you are dissatisfied with a grade you receive, you must submit your complaint briefly in writing or by email, along with supporting evidence or arguments, to me within one week of the date that I (or the TA) first attempted to return the exam or assignment to you. Complaints about grades received after the one-week deadline will be considered only if there are extraordinary circumstances for missing the deadline (e.g. student hospitalization). No new disputes will be accepted after 11:59AM two days before the course grade sheets must be turned in.

POSTED INFORMATION. The scores for tests and programming assignments will be posted on the *class web page* using a six-digit random number I will give you near the beginning of the semester.

CLASS WEB PAGE. . The class web site is accessible via the blackboard system. Course materials (e.g. the syllabus, assignments, etc.) and grades will be available via this web page, which will be the main source of current class information: (i) the daily class announcements, (ii) the programming assignments, (iii) solutions to programming assignments and exams, (iv) course reading materials, (v) TA office hours and so on.

USE OF EMAIL. You cannot expect to get last-minute help on assignments by email. More generally, you cannot expect to get detailed answers to technical questions by email. Students are encouraged to discuss important matters with the teaching team in person, typically after class or during lab or office hours. In email, include your name, and the number of the assignment or exam in question. Please include your name in the "From:" line of the email message, not just your email address. Email accounts are available free to students from the

university and commercial sources. Some commercial providers filter your email, and so you may not be receiving appended documents unless you set the permissions to do so (this is particularly true of HOTMAIL accounts). Email is a valuable tool for communicating with the teaching team. But be sure to use it properly, and follow the rules of good email etiquette (e.g. no flaming, spamming, etc.). Although it's easy for you to dash off an email question, it takes time to answer it. In general, you should not ask email questions to which you can find the answer somewhere else (e.g. class notes, web page).

OTHER COURSE RELATED POLICIES

ACADEMIC DISHONESTY (cheating): The University and the Department are committed to preserving the reputation of your UT degree. In order to guarantee that every degree means what it says it means, we must enforce a strict policy on academic honesty: Every piece of work that you turn in with your name on it must be yours and yours alone. No co-working is allowed on any test, project, or programming assignment unless explicitly allowed by me (*). As an honest student, you are responsible for enforcing this policy in three ways:

1. You must not turn in work that is not yours, except as expressly permitted by me
2. You must not enable someone else to turn in work that is not his or hers. Do not share your work with anyone else. Make sure that you adequately protect all your files. Even after you have finished a class, do not share your work or published answers with the students who come after you. They need to do their work on their own.
3. You must not allow someone to openly violate this policy because it diminishes your effort as well as that of your honest classmates.

Students who violate University rules on scholastic dishonesty in assignments or exams are subject to disciplinary penalties, including the possibility of a lowered or 0 grade on an assignment or exam, failure in the course, and/or dismissal from the University. Changing your exam answers after they have been graded, copying answers during exams, or plagiarizing the work of others (in programming assignments) will be considered academic dishonesty and will not be tolerated. Plagiarism detection software may be used on the programs submitted in this class. If cheating is discovered, a report will be made to the Dean of Students recommending a course grade of 'F' for all involved in the incident. (*) In this course when we do pair or team programming, the pair/team is treated as one individual with regard to this policy.

LEARNING DISABILITIES. If you have a learning disability that requires special attention, either during class or during an exam, please give me a letter from the Dean of Students describing what needs to be done. You should do this during the first week of classes. (The University of Texas at Austin provides upon request appropriate academic accommodations for qualified students with disabilities. For more information, contact the Office of the Dean of Students at 471-6259, 471-4641.)

RELIGIOUS HOLY DAYS. A student who is absent from an examination or cannot meet an assignment deadline due to the observance of a religious holy day may take the examination on an alternate day, submit the assignment up to 24 hours late without penalty, or be excused from the examination or assignment, if proper notice of the planned absence has been given. Proper notice must be given at least fourteen days prior to the classes scheduled on dates the student will be absent. For religious holy days that fall within the first two weeks of the semester, notice should be given on the first day of the semester. It must be personally delivered to the

instructor and signed and dated by the instructor, or sent certified mail, return receipt requested. Email notification will be accepted if received, but a student submitting such notification must receive email confirmation from the instructor. A student who fails to complete missed work within the time allowed will be subject to the normal academic penalties.

CLASSROOM BEHAVIOR. You have the right to learn in every class you attend. But you also have the responsibility to help assure that every other student shares that right. Specifically:

1. Under normal circumstances, class will start on time and end on time.
2. Come to class on time. Do not leave early. These things are very disruptive. Recognize that the buses and the parking space situation are unpredictable elements and allow for that. If you must come late or leave early (for example because of a doctor's appointment), let the instructor know in advance.
3. Don't be disruptive during class. Don't chat with your neighbors or rustle the newspaper.
4. Don't allow your electronic devices to be disruptive. Turn off your cell phone, beeper, and watch alarm.
5. Don't leave your mess lying on the classroom floor when you leave – pick it up and throw it in a trash can.

EXTERNAL TUTORING. For those students having considerable difficulties with the course material, individual tutoring is provided by certain organizations not directly affiliated with this course. See the following references:

1. Eta Kappa Nu – they will announce their tutoring schedule shortly after the semester begins
2. The Learning Skills Center, located in Jester A332A, 471-3614, has individual tutors for hire. The cost is about $10 per hour; students receiving financial aid can get 2 hours per week free. See the web page http://www.utexas.edu/student/utlc/tap.html for more information.

DISCLAIMER. I occasionally tell jokes and stories during class as a way of breaking up the technical material that we're covering. A story might be a simple observation about campus life, about something that has happened to me as a student or a professor, or it may even be an aggie joke. These stories may reflect a point of view that is different from your own. Hopefully, most will be interesting or funny. Some, almost certainly, will be politically incorrect. None of these are intended to be offensive.

COURSE POLICIES CAVEAT. As departmental, college and UT policies change, I reserve the right to alter the effected course policies stated herein during the course of the semester.