Intro to CS II with Intensive C++
CAS CS-113 – Spring 2003
http://www.cs.bu.edu/fac/byers/113/cs113.html

TR 9:30 - 11:00 AM, CAS 318

	Instructor: Prof. John Byers	**Teaching Fellow: Marwan Fayed**
Office	MCS 280, Phone: 353-8925	MCS B24
Email	byers@cs.bu.edu	mfayed@cs.bu.edu
Web	www.cs.bu.edu/fac/byers	www.cs-people.bu.edu/mfayed
Office Hours	M 1:30 - 3:00 & Th 11:00 - 12:30	M 3:00 - 4:00 & W 4:30 - 6:30

Course Overview: This course begins with an intensive introduction to C++ and then covers all the materials in CS 112. You will learn advanced programming techniques involving dynamic memory allocation, pointers, linked lists, stacks, recursion, trees, and some searching and sorting. All of this will be embedded into the highly-disciplined structure provided by the object-oriented programming language C++.

Prerequisites: This course is designed for students who already program proficiently in PASCAL, C, Java, or some other high-level programming language. Note: If you do not have such previous exposure to programming, then you are requested to transfer to CS 111. You are expected to be familiar with UNIX and EMACS (or other equivalent text editor). Some help will be available in the section, but if you have not used UNIX or EMACS before, then you should attend the appropriate tutorials provided by B.U. Office of Information Technology: http://www.bu.edu/cc/tutorials/.

Textbooks: The required textbooks are:

1. Deitel and Deitel. C++ How to Program. Prentice-Hall, 2003, ISBN 0-13-038474-7.
2. F. Carrano and J. Pritchard. Data Abstraction and Problem Solving with C++, Addison-Wesley, 2002, ISBN 0-201-74119-9.

Topics: We will no doubt drift from any formalized plan, but a rough schedule of where we are headed is provided in the pages that follow. A more detailed and continually updated schedule will be maintained on the course homepage.

Workload: Be forewarned – the workload in this course will be heavy. To master the conceptual material covered in lecture and to become a strong C++ programmer, there will be substantial programming assignments due approximately every other week. This is an intensive course that covers the topics of CS 111 and CS 112 in one semester.

Grading: The course grade will break down as follows:

- 40% programming projects
- 20% midterm
- 30% comprehensive final
- 10% labs, attendance and class participation

The average course grade for past offerings of this course is "B-" and incompletes will not be granted.

Exams: There will be a ninety minute in-class midterm held during the middle of the semester, likely Thursday, February 27. The final will be held during the normal final exam slot. Please make your end-of-semester travel plans accordingly. In the event of serious illness documented by a doctor's note, makeup examinations will be given orally.

Program Submissions Programming assignments will be submitted via the gsubmit program, usage of which is documented on the course homepage and will be discussed in class. All assignments will be tested for originality by an automated software tool.

Attendance: It is expected that you will attend lecture and the laboratory section for this course and I will take attendance at the beginning of lecture. I strongly encourage you to arrive in class on time, since it is highly disruptive to have students flowing in throughout the class period. Moreover, when students are at a borderline between grades, I will check the attendance records before making a final determination.

Late Policy: Programming assignments are typically due Mondays at 10PM. During the course, you will have **two** opportunities to turn in an assignment up to 24 hours late with no penalty. Under no circumstances will additional time be granted, nor will additional late submissions be granted.

CAS Academic Conduct Code: Academic standards and the code of academic conduct are taken very seriously by our university, the College of Arts and Sciences, and the Department of Computer Science. Course participants must adhere to the CAS Academic Conduct Code – please take the time to review this document if you are unfamiliar with its contents. All instances of academic dishonesty will be reported to the academic conduct committee; first time violators are routinely suspended for a semester or more.

Collaboration Policy: The work that you submit must be your own original work and it is an act of plagiarism to represent the work of another as your own. You are encouraged to discuss the general nature of solutions with other students in the course, but it is not acceptable to collaborate in writing lines of code, nor to share or copy code. Any discussion or collaboration with other students in the course must also be acknowledged in your submission. If you are uncertain whether an action constitutes a violation of the collaboration policy, I will be glad to discuss the matter with you.