1 Course Objectives

This course has five main objectives:

1. **abstraction.** You should understand the importance of abstraction in the construction of computer systems, and be able to exploit both data abstraction and procedural abstraction.
2. **basic techniques.** Recursion, pointers, and functional parameters are three basic programming tools. You should become thoroughly familiar with each.
3. **structures and algorithms.** You should learn the particulars of four commonly–used abstract data types (lists, stacks, queues, and binary trees) and two classes of algorithms (searching and sorting).
4. **practice.** Practice is essential for the development of coding skills. You should gain significant coding experience by programming the homeworks and projects.
5. **style.** You should learn the characteristics of good programming style, and begin to develop your own style.

2 Times and Locations

Class meets Tuesdays and Thursdays 10:00–11:15 a.m. in Lecture Hall 4 (the large lecture hall in the Academic IV building).

Section meets Mondays and Wednesdays (your choice) in CP208 from 10:00 a.m. until 10:50 a.m.

The Honors section meets Wednesdays from 10:00 a.m. until 10:50 a.m. in MP103.

3 Course Staff

3.1 Instructor

Jim Mayfield

Phone:	(410) 455–3099
Office:	ECS room 225H
Email:	mayfield@cs
Office Hours:	Tuesday, Thursday 2:00–3:00 p.m. or by appointment

3.2 Teaching Assistant

Venkat Krishnamoorthi

Phone:	(410) 455–2861
Office:	ECS room 231E
Email:	venkat@cs
Office Hours:	Monday, Wednesday 11:00–12:00 a.m.
	or by appointment

4 Textbook

The course text is *A Book on C* (third edition) by Al Kelley and Ira Pohl, Benjamin Cummings, 1995. The text is available at the campus bookstore. Assigned readings for each class are listed in the Syllabus; you are responsible for keeping up with these readings.

5 Grading

Course grades will be based on the following work:

5 Homeworks	@ 4%	= 20%
3 Projects	@ 8%	= 24%
Section attendance		= 6%
2 Midterms	@ 15%	= 30%
Final		= 20%

As per University policy, incompletes will be granted only under extraordinary circumstances; students who are enrolled April 12 (the last day to drop a class) should be prepared to receive a grade of A–F.

5.1 Projects and Homework

There will be three projects and five homeworks. Each assignment will have a due date. Assignments that are submitted by midnight on the due date will receive an on–time bonus. Each assignment carries a two–day grace period after the due date, during which assignments may be turned in with no penalty. No assignment will be accepted more than two days late (*i.e.*, after midnight on the day the grace period expires); make sure that you submit any work you have done by this time. Bear in mind that assignments are due on the due date; the automatic extension is provided for your convenience and to ameliorate unforseen problems (such as machine failures). You should complete and submit your project by the due date, then resubmit your work if you make updates during the grace period.

To receive more than token credit, a program *must compile and run*. Grading will be based on the following criteria:

Design (including algorithm, data structures, modularity, *etc.*)
Implementation (including correctness, robustness, *etc.*)
Documentation (including comments, layout, *etc.*)

Venkat Krishnamoorthi, the course TA, will discuss these grading criteria in more detail in one of the sections.

In addition to the on–time bonus, assignments that are particularly well–conceived and executed may receive extra credit at the discretion of the course staff. Bonus points and extra credit will be used to determine which borderline grades are moved upward. Thus, lack of such points cannot hurt your grade (*e.g.*, you can get an A without them), but having them can help your grade.

Accounts will be available on the gl machines. Your account will allow you to use both terminals and the SGI workstations in ECS rooms 005, 020, and 104. You need not develop your programs on the gl machines, but you must be able to upload and run them there. Be aware that these machines are often overcrowded just before assignments are due; your best bet is to get the assignments done early.

Many students will already have a computer account on the UMBC systems. If you do not yet have a computer account, go to any UCS laboratory that has SGI workstations (ECS 005, 020, or 104), login as *register*, and follow the instructions (you can also telnet to gl and login as *register*). Your account will then be available for pickup in ECS 019. You are issued a single account for all your classes, and you keep this account as long as you are registered at the University. If you need help in obtaining your account, see the Help Consultant in ECS 005.

5.2 Sections

Weekly section attendance is mandatory, although you may attend either the Monday or the Wednesday section. Attendance at sections is worth 6% of your grade. You may miss up to three sections with no penalty. At some sections you will take a quiz. The quiz will be graded immediately, thereby giving you feedback on how well you understand the week's material. The grades you receive on the quizzes *will not count toward your final grade;* they are intended only to give both you and the course staff feedback on the week's topics.

5.3 Exams

There will be two midterms and a final. The material covered by the exams will be drawn from lectures, from handouts, and from the projects and homeworks. Each exam will be cumulative, but will emphasize the material covered since the previous exam.

6 Handouts and On–Line Documentation

There will be periodic handouts during the semester (starting with this Prospectus), containing course information, assignments, and discussion of important topics. Handouts will be distributed during lecture. I will not typically bring copies of past handouts to lecture; such handouts will be available on–line via a distributed hypertext system called the World Wide Web (WWW). The easiest way to access this information is through the NetScape program. NetScape can be run from any SGI workstation using '`netscape ~mayfield/202.html`'. Upon executing this command, you will be presented with a screen of basic information. Some entries will be underlined; such entries represent connections to other screens of information. To travel to another screen of information, simply click the mouse over an underlined entry. To return to the previous screen, click on the entry labeled 'back' at the top of the window. For more information about WWW and NetScape, look under the NetScape 'Directory' menu, or use the `man` command. If you are using a terminal (as opposed to a workstation), you should use the 'lynx' program instead of NetScape. 'XMosaic' is another alternative to NetScape.

All course handouts will be available on–line through NetScape in both WWW and printable forms. If you miss or lose a handout, you should obtain it on–line. Printable versions of handouts are said to be in *PostScript* format. PostScript may be viewed using the *ghostview* program, or printed directly to a laser printer. For more information on ghostview, use '`man ghostview`.'

7 Computer Science Help Center

The *Computer Science Help Center* (CSHC) provides tutoring help on homework, projects, exams, and general concepts. The CSHC is located in ECS 334A. Hours of operation were unavailable at press time. Email may be addressed to `cshc@cs`. The CSHC is designed not to replace course staff office hours, but to augment them; feel free to take advantage of both.

8 Academic Honesty

Cheating in any form will not be tolerated. In particular, examinations are to be your own work—not your neighbor's and not your crib sheet's. You may discuss the programming assignments with anyone. However, any help you receive *must* be documented. At the beginning of your program, you must include a comment indicating the sources you used while working on the assignment (excluding course staff, consultants' desk, CSHC, and textbook), and the type of help you received from them. If you received no help, say so. *Failure to include this comment at the top of your program will result in your assignment being returned ungraded.*

The implementation of the assignments must be your own work. If you are stumped on a particular error, you may consult with someone else; however, if you consult with someone other than the instructor, the TA, the CSHC, or the consultants' desk, you must place a comment in your code near the point of the error, stating the source and scope of the help you received. Reasonable help will not affect your grade; failure to cite your sources is academically dishonest, and will be dealt with harshly.

Each semester, students fail the class because they do not write their own programs. Do not fall into this trap. Neither borrower nor lender of code be. The only way to become a good coder is to practice coding. If you copy code from another student, you gain as much coding experience as an art student gains drawing experience by photocopying someone else's picture. If you give your code to another student, you do him or her a disservice (in addition to setting yourself up for a failing grade). If you are having difficulty with a homework or project *see the course staff.* The earlier you come to us, the more help we can give you. However, even an eleventh hour visit to the course staff is preferable to 'borrowing' code.