Chapter 1

Course Information

1.1 Course Guidelines

1.1.1 Grading

Your grade for this course will be based on your numeric score on the homework assignments, hourlies, and the final exam.

The grades will be weighted as follows:

- 50% - five homework assignments (10% each).

  Note that there is an optional sixth assignment; see below for more information.

- 20% - two hourly tests (10% each).

- 30% - one final exam.

The hourlies and the exam are **closed book**: no notes, computers, calculators, or other aids will be permitted. Relevant definitions and formulas will be provided for you on the test.

Assignments represent individual work; no collaboration is permitted. All questions about the assignments should be directed to the course staff; discussion between students about the specific problems on the assignments is not allowed. (Discussion about the course material, however, is strongly encouraged. Study groups are a wonderful way to master the material.)

An optional sixth "make-up" assignment will be made available to any students who specifically request it. The grade for this assignment will replace the lowest grade of the five ordinary assignments, and provides an opportunity for students who do poorly on one of the assignments to demonstrate that they have mastered the material and improve their grade. Note that it is possible that doing the "make-up" assignment will *lower* your grade– if your score on the sixth assignment is lower than the lowest score of your other five assignments, then your cumulative score will also be lower!

1.1.2 Lateness Policy

The course works best when homework is handed in on time– it can be graded and returned promptly (giving you valuable feedback for your next assignment), solution sets can be distributed, and I can get a feel for how well you have learned the material. However, I realize that sometimes this is not possible, due to other commitments you might have, or personal emergencies that might arise.

- Requests for an extension should be made directly to the professor, either in person or via email (ellard@deas.harvard.edu) or telephone (496-6434 during the day, 643-9644 at night). Except in the case of extreme emergencies, requests for an extension must be made at or before the time that the assignment is due.
- There is a deduction of 10% per day late. Late days are not subdivisible; one minute late is the same as one day.
- There is a "grace period" of three days that can be used at any time during the semester. This can be subdivided among two or three assignments, or used all at once. Simply send email to your TF when you submit your assignment, saying that you are using your grace days, and the lateness penalty will not be taken. Once again, late days are not subdivisible; one minute late is the same as an entire day.
- The first two assignments (which are due on Mondays) have a drop-dead date of the following Thursday. The other assignments, which are due on Thursdays, have a drop-dead date of the following Monday. Homework submitted after this time will receive a grade of zero, unless an extension has already been granted.

  Note that the drop-dead dates mentioned earlier are not effected by the grace days.

1.2 Homework Guidelines

In my opinion, the homework assignments are the most important aspect of the course. They are time-consuming and challenging, and should never be left until the last minute.

I have designed the homework assignments to be difficult, but manageable. Some of the problems are straightforward, but others require considerable thought and perhaps a touch of inspiration to solve. I make mistakes, however– sometimes an assignment is much harder or easier than I have anticipated! If you find yourself struggling for days on a single problem, please let me know so I can clarify the problem, restate it, or possibly omit it. At the same time, if you find that there are other problems that you can do effortlessly in a few minutes, let me know so I can challenge you with problems you will find more interesting and educational.

1.2.1 How To Approach the Assignments

- **Start early, and pace yourself.** If you wait until the last minute to begin the assignments, you will regret it.

  The assignments are time-consuming, but not all of that time is productively spent in front of a computer. I personally find that it is most effective for me to read over the assignment as soon as possible, and then ponder each over a period of several days, until all the pieces from lecture and the readings fall into place and I've got a good idea how to solve them, and *then* start coding.

  Starting early also gives you an opportunity to discover that you might need a little extra help from me or a TF to understand some topic related to the assignment.

- **There are no trick questions.**

  If something seems too hard (or too easy), ask about it. There are no *intentionally* misleading or tricky questions, Please let me know if you find any, so I can clarify them.

- **Don't Get Stuck On One Problem.**

  Don't spend all your time working on one very hard problem. Along the same lines, don't spend all of your time gold-plating your answers to the easier problems.

- **Pay Attention To Feedback.**

  In addition to your grade, your TF will give you feedback about your work. Word for word, these are probably the most important things you will read all semester– so study them carefully!

1.2.2 Written Exercises

The assignments usually include several questions that do not require any coding, but instead require written answers. These questions specify the name of the file where you should write your answers.

Please write your answers in clear English or pseudo-code, with tables or diagrams when appropriate or requested. Use complete sentences.

Ordinary text and legible hand-drawn diagrams are sufficient– please do not feel that it is necessary to spend time and energy making your answers into works of art.

1.2.3 Coding Guidelines

1. **Put your name in a comment at the start of each file you edit or change in any way.** In addition, please staple or fasten together securely all the pages of your printouts.

2. **All code must be written in ANSI C, and must compile successfully. Non-compiling code will not be accepted.**

   At this point in your programming career, I don't expect this to be a problem.

3. **All code must compile and run properly on the course (HP-UX) machines.**

   We will *test* and grade your assignments on the course HP-UX machines, so it is in your best interest to test your programs there as well. Code that doesn't work on the HPs is wrong, whether or not it works properly anywhere else.

   In general, I don't care what platform you do your programming on. For some problems, however, we'll give you libraries or sample executables, and unfortunately these libraries and programs usually function properly only on the course machines. For these problems, you *must* use the course machines to compile and test your code.

4. **Don't change the code that is marked** *DO NOT CHANGE.*

   Some of the code that we'll give you should not be changed. Don't change it.

   The code we want you to write or edit should be clearly marked. Look for instructions like **WRITE THIS FUNCTION** in the code or consult the assignment (and when in doubt, ask).

5. **Code to the specifications given, not to the apparent requirements.**

You may notice that, in some cases, some functionality of the code that we ask you to write doesn't seem to ever actually be used by the examples or sample programs– but taking advantage of this fact (either implicitly or explicitly) is a very bad idea, because we grade against the spec.

6. **We do not grade on minor style considerations.** I do reserve the right to deduct points for truly awful code, however.

   In many programming courses, there are deductions for code that doesn't match specific consistency and style guidelines. This course does not have any specific guidelines, except that the grader should be able to figure out how your code works by reading it– it should be abundantly obvious from the organization of the code and the comments you write.