operating systems, for example) is important, because many of the design decisions made in system software are motivated by these problems. This course, as a project course, is part of your general training in this aspect of computer science and engineering. The project and our class discussions may not always be related; it is your responsibility to manage <u>both</u> parts of this class.

There will be a midterm test and a final exam.

Software:

You are expected to master the tools you need to complete these assignments (e.g., the web, Unix, dbx debugger, C, C++, X, make, RCS, etc.) basically on your own. This does not mean that you shouldn't come to us with questions, but you will be happiest if you make normal efforts to figure things out for yourself first. (Try 'man', obviously, and perhaps a little experimentation.)

Evaluation:

Classroom participation will be worth roughly 5% of your grade, the two exams roughly 55%, and the assignments and project roughly 40%.

It is always a problem assigning grades to the individual members of a team. Our policy is that the default is to give each project member the same grade. However, if you feel that you or someone else in your group will be treated unfairly under this scheme, talk to us as soon as poosible, but in all cases before the project has been graded.

Rules of the Road:

The assignments, especially the project assignments, are incremental. If you fall behind you'll either be permanently lost or else expending superhuman effort trying to catch up.

Homeworks that are more than one week late will not be accepted. You may hand in one assignment late (up to a week) without penalty. After that, there will be an automatic 25% penalty for late assignments (again, up to a week). The ‘‘free’’ late assignment is designed to minimize the trouble to everyone in presenting legitimate excuses. You should assume that there is no such thing as a legitimate need to hand in multiple late assignments.

Please use good judgement concerning collaboration. What you get out of the course will be directly proportional to what you personally put into it.

CSE 451: Introduction to Operating Systems

Winter 2000

Instructor:

Hank Levy
Sieg Hall 412
543-9204
levy@cs.washington.edu

Office Hours: M/W 10:30 - 11:20

Please don't feel shy about coming to office hours with any questions you might have. For one thing, it helps me to find out what hasn't been clear from the reading and the lectures. For another, I just might be able to answer your question.

TAs:

Mike Swift
mikesw@cs.washington.edu

Tim James
tjames@cs.washington.edu

TA Office Hours: to be determined

Text: Silberschatz and Galvin, *Operating System Concepts*, Addison-Wesley.

Course Organization:

There will be three kinds of assignments: reading (primarily from the book), questions from the book, and (incremental) a multi-person project (i.e., programming).

Assignments will be weekly. You are basically responsible for reading the appropriate sections of the book. Please please please read these sections in advance of our lecture discussions. The assigned questions are intended to help you verify that you've understood the important parts of the reading. You should expect that not all of them will be graded by us.

I will give out a description of the project shortly. The size of the implementation effort and the requirement to work in teams are both explicitly part of the goals of these assignments. Understanding the problems associated with building and maintaining large pieces of software (like