DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.033—Computer System Engineering Handout 2—February 6, 1996

ASSIGNMENT: February 6 through February 13

Each week you can expect to find an assignment like this one on the 6.033 home page on the Web, telling you what you should read over the course of the following week. Occasionally we will be organized enough to predict the readings a few days beyond the next Tuesday. Numbered readings, such as "reading #2," refer to materials in the packet you pick up from the EECS Instrument Room (38-501).

Each week you are expected to complete a reading report. The reading report is a short (one page) written report due at the beginning of every Tuesday recitation. The specific topic to be addressed in the report will be given on the previous week's assignment sheet. In recitation you should be preparedto talk about the whole paper, not just the particular point of the writing assignment.

For Lecture: Wednesday, February 7

Read Saltzer, Chapter 1 (reading #3), and begin reading the paper by Simon, "The Architecture of Complexity" (reading #4).

For Recitation: Thursday, February 8

Get online and find the 6.033 handouts on the Web. We are on the brink of a revolution in publishing. We expect that electronic publishing (as opposed to paper distribution) will happen in the next decade. To be part of this revolution and to experiment with it, 6.033 will attempt to avoid paper publishing and publish all handouts electronically. For more information about on-line browsing see Handout 1.

Finish reading Simon, "The Architecture of Complexity" (reading #4). Simon discusses the power of hierarchy using an astonishing variety of examples from many fields. As a personal exercise, after reading the paper, you may want to identify two examples of the use of hierarchy in Project Athena, and understand how hierarchy provided significant leverage or advantage.

In addition, read Gabriel's paper (reading #2). This is a lightweight paper, but makes a thought-provoking point. In general, we will have at most one heavy-duty technical paper per recitation, sometimes accompanied by an easy-to-read paper with up-to-date information, a thought-provoking point, an issue important to society, or high entertainment value.

For Special Lecture, Friday, February 9

This lecture is provided by the staff of the Writing Program, and is intended to help you do a better job on weekly one-page reports. In preparation for this lecture, read Gopen and Swan, "The Science of Scientific Writing" (reading #5).

For Lecture, Monday, February 12

Read Saltzer, Chapter 2 (reading #6).

For Recitation, Tuesday, February 13

Read the Leveson paper (reading #7) and write a one-page reading report that addresses the following question.

> On page 39 (bottom of the third column), Leveson and Turner state that code reuse often leads to safety problems. "Rewriting the entire software to get a clean and simple design may be safer in many cases." What problems can code reuse lead to? Does this contradict conventional wisdom and software engineering practice regarding code reuse? Discuss.

Check out the 1995 6.033 web pages for examples of good writing on one-page reading reports. One URL is http://web.mit.edu/6.033/1995/www/reading.html. Also, please check the 6.033 FAQ for formatting instructions. The FAQ is available through the 6.033 Home Page.

After reading Leveson's paper you may be flabbergasted by how badly the Therac-25 systems were designed. The next reading (#8, Lampson's "Hints for computer system design") has a number of suggestions that often lead to better systems. If you have time and energy left, browse through this thoughtful paper. You might want to ask yourself which hints the designers of Therac-25 violated. Don't worry if you do not fully understand this paper yet (or if you only have time to skim it); it will be assigned again at the end of the term.

System aphorism of the week

There is no such thing as a small change in a large system.