Cop 4020 Programing Languages — Instructor Bellenot
Fall 1989 — Office 218 Love
Office Hours: MWF 12:45 - 1:15 & by appointment

Texts: MacLennan, Principles of Programming Languages, 2nd Ed. 1987 (main text) and Kernighan and Ritchie, The C Programming Language, 2nd Ed., Prentice Hall, 1988.

Coverage: We will attempt to cover the entire main text.

Grades: If both your test average and your programing average are at least a C-, then your grade will be the average of these two grades, else your grade will be the minimum of these two grades.

Tests: Final at the regularly scheduled time (which might be Friday 15 Dec. 10 am -12 noon). Midterm tentativity scheduled for 25 October. Tentative Quiz dates 22 Sept., 6 & 20 Oct., 8 Nov. and 1 Dec. There will be NO make-up tests or quizes. Your QA (quiz average) is the average of your four best quizes. Your test average is (QA + MidTerm + 2* Final)/4.

Programing Projects: Most of the programs will be written in "ANSI-C". The big project will be developing a Lisp Interpreter in C. The first program will be a Lisp program and if time permits the last program will be in C++. Other programs will be assigned. Programing will be done on the department VAX (named "fsucs") which runs Ultrix (DEC's UNIX).

Program Grades: The following guidelines will be used (in spirit):

A (100-90): Correct and well written in a clear and economical style. (This applies to documentation as well as code.)

B (89-80): Program works on public and private test data and indicates understanding of all the principles behind the assignment.

C (79-70): Program works on public test data, is based on correct algorithms and shows understanding of most the principles of the assignment.

D (69-60): Program tuned in and shows some understanding of the assignment and works on some of the public test data.

F (59-0): Program not turned in, copied or incomplete.

Murphy's Law

My Programming Languages class suffered through many technical difficulties which were frustrating. Although the VAX (fsucs) had successfully used in programming languages last fall, the computer science department decided to change the way that students accessed the VAX. (They shut down the terminal room that was near where the VAX is physically located.)

1. The students accounts were set up as if they were logging in on the old terminals, however they were logging in over the network. UNIX requires the user tell the system what kind of terminal he/she is using. If you fail to state what kind you have, then it sets your terminal to "dumb" and no "screen editor" works as advertised with a dumb terminal. Figuring out how to set up your terminal in UNIX is non-trivial.

2. The first work-around to 1 above didn't make best use of the keyboards in the PClab, requiring a second work-around.

3. Next the telnet software which the students were using to go from the PClab to the VAX was "upgraded". As a result of this upgrade, the connection between the PClab and the VAX was "broken". The next work-around required them to go from the PClab to Omicron (a sun file server like Gauss) and log in as a phony user named "fsucs" which then connected them to the VAX.

4. A combination of the VAX , the editor emacs and this telnet caused control flow characters ^S and ^Q to get mistaken for editor commands. This wasn't happening to all students nor was it happening all the time. Again there was a fix, actually two fixes since the ^Q problem was discovered after the ^S problem was fixed. (^Sxyz<CR> is the command to search for the string "xyz"; <CR> stands for the return key. ^Qx is the command to insert the next key "x" into the file, no matter what; this is useful for putting control characters into files.)

5. A combination of knowing too little about UNIX allowed some of them to shoot their foot. Assignments were turned in by mail using the command "mail bellenot < file-to-be-turned-in" which sends the file to me. For some reason about the time the first big assignment was due a bunch of students forgot if it was < or > and guessed. The command "mail bellenot > file-to-be-turned-in" immediately

deletes the file "file-to-be-turned-in" and creates a few lines of mail header stuff in a new file called "file-to-be-turned-in" and hangs waiting for input. Eventually, the student types either two ^C's which kills the letter or the student types ^D which sends a void letter to me. In either case the file "file-to-be-turned-in" is ruined and contains what looks like a mail message. Some students realized that they had guessed the wrong one and so now they used the correct "mail bellenot < file-to-be-turned-in" sending me the trashed file. (One student decided to edit this trashed file. He decided the editor was broken since it seemed to put him in mail every time he edited the file.) Fortunately the editor emacs makes backup copies.

6. Just to make things more fun, Murphy struck the hardware too. The VAX was real flaky during the first few weeks of class. Sometimes it went down more than once a day.

All these problems (accept 5) had been corrected before the first major programming assignment. But it was frustrating to both to me and the students.