UNIVERSITY of CALIFORNIA at Berkeley
Department of Electrical Engineering and Computer Sciences
Computer Sciences Division

CS61C
Course Info

Kurt Meinz
Summer 2004

CS61C: Machine Structures
General Course Information

1 Introduction

The CS 61 series is an introduction to computer science, with particular emphasis on software and on machines from a programmer's point of view. The first course, CS 61A, concentrated mostly on the idea of *abstraction*, allowing the programmer to think in terms appropriate to the problem rather than in low-level operations dictated by the computer hardware. The next course, CS 61B, dealt with the more advanced engineering aspects of software—on constructing and analyzing large programs and on techniques for handling computationally expensive programs. This course, the last in the series, concentrates on machines and how they carry out the programs you write.

In CS 61C, we are interested in giving you an introduction to the interfaces and operations of modern microprocessors. At the beginning of the course, we will introduce you to the low-level programming language *C* and the *MIPS* assembly and machine languages. A fourth language, the hardware description language *Verilog*, will be used as a basis for describing machine fundamentals such as *caches*, *virtual memory*, *pipelining*, *gate-level digital design*, and *input/output*.

2 Do You Belong Here?

The summer session version of this course is a bit different from the regular semester version. We cover all of the usual material, but we do it in **half** the time. This makes the course *very fast*. If you fall behind, you will find it almost impossible to catch up. At the same time, the summer course has no restrictions on enrollment. During the regular semester, CS 61B is a prerequisite for this course. However, over the summer, we will waive this requirement. Therefore, anyone, regardless of prior experience, may enroll in the course (until it fills.) We encourage anyone who's curious or interested to take this course, even if they aren't computer science majors!

With that said, this course will be difficult and time-consuming. Your nominal classroom hours are roughly 12hrs/week – however, you can expect to be spending, at the very least, another 20hrs/week on readings and assignments. If you have other time commitments, such as a summer job or another summer course, you may find yourself stretched too thin. In short, this course will be like a full-time job, so please plan accordingly.

3 Course Materials

The textbooks for this course is *Computer Organization and Design: The Hardware/Software Interface* (hereafter *COD*) by Patterson and Hennessey, second edition, and *The C Programming Language* (hereafter *K/R*) by Kernighan and Ritchie, second edition. Both should be available in the textbook section of the ASUC bookstore and other local textbook sellers. **You must get the second editions! Don't buy a used copies of the first editions.** If you cannot afford or don't want to buy the books, copies of them are on reserve at the Engineering Library.

If you haven't used Unix before, you might also consider the *User's Guide to Unix and the EECS Instructional Facilities* available at CopyCentral, 2483 Hearst Avenue (x Euclid). We will, however, make an effort to link to Unix help documents from the course webpage.

4 Enrollment—Laboratory and Discussion Sections

Summer session is 8 weeks, with every week packing in two standard course weeks. This course is normally structured so that there is one discussion and one lab meeting each week; but we must pack both meetings into the first two days of the week, and again, both into the last two days of the week. Generally, the lab portion occurs some time between Monday and Tuesday's lectures and again between Wednesday and Thursday's lecture. The discussion sections meet between Tuesday and Wednesday's lecture and again between Thursday and the next Monday's. You will also need to spend additional time working on the computers in the Soda Hall labs. Most weeks, the first and third meetings will be in our discussion room, 320 Soda Hall; the second and fourth meetings will be in the laboratory, 271 Soda Hall.

The discussion and lab sections are run by Teaching Assistants; each TA will handle enrollment and grading for his or her sections. We anticipate some rearrangements during the first week in response to oversubscribed or undersubscribed sections. **If you are waitlisted or your section has been cancelled,** you should communicate via email with the TA who is in charge of the sections that **you would like to move into** but be prepared to be flexible if your first choice is full. Please be in a definite discussion section by the end of this week.

You must have a computer account on the 61C course facility. You must set up your account *before Noon on Wednesday, June 23* because that is how we know who is really in the class. Account forms will be distributed in the LAB SECTIONS. The first time you log in, you will be asked to type in your name, SID (if you have one), and a secret code. The secret code will be used to retrieve your grades from Webgrades. Remember the code, and, for security purposes, **make it distinct from your account password!**

Please follow the registration instructions carefully. You must get your account *and log into it* no later than **12:01 PM Wednesday** so that we have an accurate class count. To reiterate: Everyone MUST log in by Wednesday Noon (or have made special arrangements with their TA) **OR YOU WILL BE DROPPED** from the course!

Some of you have personal computers and may want to do the course work at home. This is fine with us, although you'll have to be careful to install proper software on your home computer to make your computer behave like the ones we use in the labs. In any case, though, you must get a class account even if you intend never to use it.

5 How to get the most from this course

We recognize that everyone's style of learning is unique. Some students are excellent at studying–they work hard, and are extremely diligent. They do all the readings conscientiously, and work all the problems. Some students are incredibly quick, and get by doing little of the reading, even less of the homework, and still ace the tests. Some students learn best by listening to lecture, and discussing it with their friends and TAs. Some students are aiming for the A+, others just to get by with a passing grade. Usually, students are some of each of these types, or are sometimes one, sometimes another. Since everyone's style is their own, we try to provide as many opportunities to learn this material as possible. Therefore, use them all, and learn what works best for you.

Different people solve problems differently; there are often many right answers to the problems in this course. And of course, what you find easy, your friend may find hard, and vice versa. Therefore, the best way to learn is to talk with other people, and ask them questions when you are stuck. Even if you think you understand everything, you will learn the material better if you have to try to explain it to someone else. In addition, learning how to think about the problems in many different ways will solidify your understanding of this material.

Finally, is it possible that some of you feel uncomfortable telling others when you don't understand something. Many of us find it hard to ask questions–all the more reason to overcome this fear early! The ability to ask for help is a wonderful strength that will serve you well in life. Throughout this course, we will try to encourage you to ask each other, and the TAs and myself for help.

6 Information Resources

The Teaching Assistants who teach the discussion sections are also available to answer questions. You may drop in during office hours, make appointments for other times, or communicate with them by electronic mail. Feel free to visit any of the TAs–not just your own! You may find that hearing different people's explanations helps if at first you do not understand some material.

For technical questions about the homework or projects, or administrative questions such as missing homework grades, send electronic mail to your particular TA or reader. You can also send mail about intellectual questions to me, but if it's about grades I'll just refer you to your TA.

In addition, there is an electronic bulletin board system that you can use to communicate with other 61C students and staff. The ucb newsgroup can be read only from machines in the berkeley.edu domain, so if your net connection is though a commercial ISP then you must log into a lab machine to read the newsgroup or try this:

`http://www-inst.eecs.berkeley.edu/connecting.html`

Please do not send electronic mail to every student individually! That would waste a lot of disk space, even for a small message. Use the newsgroup instead. Electronic mail is for messages to individuals, not to groups.

There is a class web page, with online versions of some of the documents we hand out:

`http://www-inst.eecs.berkeley.edu/~cs61c`

Tutoring services are provided by Eta Kappa Nu (HKN), the EECS honors society, and Upsilon Pi Epsilon, the Computer Science honors society. They share an office in 345 Soda; call them at 2-9952 or send e-mail to `hkn@hkn` or to `upe@cory`. The Student Learning Center may also provide tutoring hours this summer. I will post a link to the SLC in the webpage.

Additional information to help you in studying, including hints from the course staff and copies of programs demonstrated in lectures, is available at the course website.

7 Computer Resources

The computing laboratory in 271 Soda Hall consists of about 35 SunRay terminals connected to a Sun Solaris server. This is our primary lab room, although the CS 61C accounts can also be used from any EECS Instructional lab in Soda or Cory Hall.

The lab in 271 Soda Hall is normally available for use at all times, but **you need a card key for afterhours access to the lab**; to get a card key, stop by the 3rd floor office of Soda Hall and fill out a form for a card key. You will need a $20 deposit to get the card key. The card key will give you access to the 2nd and 3rd floors of Soda Hall so that you may enter at any time, day or night. Do this today! During scheduled lab sessions, only students enrolled in that particular section may be in the lab. Since lab sections run from early morning until late evening, you might need to use the other Soda Hall labs to work on homework outside of class. In particular, 273 Soda Hall should be at your disposal at all times. When sections are not in session, any 61A student may use any of the 2nd floor labs on a drop-in basis. If there are no free workstations, please feel free to ask anyone who is not doing course work to leave. In particular, *game playing is not permitted*. We are relying on social pressure to discourage abuse (such as stealing the chairs or monopolizing a workstation for six hours during prime time to play chess). Therefore, do not feel embarrassed to apply such pressure.

These machines use the Unix operating system, a timesharing system that is quite different from the microcomputer systems you have probably seen elsewhere. The course reader includes introductory docu-

mentation about Unix and about Emacs, the text editing program we are recommending for your use. (It is one of several Unix text editors; you'll find that everyone has his or her own favorite editor and hates all the others.) Although the use of Unix is not taught in 61C lectures, it will be extremely worthwhile for you to spend some time getting to know how the system works.

The Computer Science Undergraduate Association (CSUA), Open Computing Facility (OCF), and Experimental Computing Facility (XCF) usually offer introductory Unix training sessions. Details will be announced when we have them.

If you have a home computer and a modem, you may wish to use your class account remotely. If so, you are encouraged to use a commercial Internet Service Provider to connect to the campus; several companies offer student rates. Again, check out

`http://www-inst.eecs.berkeley.edu/connecting.html`

8 Computer Community Spirit

If you have lived in a dorm or other concentrated student housing, you have already learned that any facility shared by a large group of people is fertile ground for practical jokes. You've also learned that selfishness in the use of common facilities can lead to a lot of *negative energy*. Computers are no different. For example, there is only a finite amount of file storage space. If you fill it up with digitized pictures of all your friends, other people can't get their homework done.

In the dorm, people generally have a good sense of perspective about what's funny and what isn't. Filling up your friend's room across the hall with balloons is funny. Filling it up with water balloons or live crickets or a 400 pound toilet is on the edge. Filling it up with epoxy isn't funny at all. But, for some reason, some people seem to lose that sense of perspective when it comes to computers. Perhaps it's because the damaged property is intangible; perhaps it's because with a computer you don't have to be physically near the victim. Whatever the reason, try to overcome it. It's not funny if someone can't complete the course work because you deleted their files.

The operating system we use provides enough security so that nothing you do will mess up another user by accident if you're minding your own business. It is certainly possible to mess up the system deliberately. Many of you are familiar with the personal computer environment, in which some people consider it a mark of sophistication to write "virus" programs that interfere with other people's computers. You are now entering a different culture with different values. Our research work, as at any university, depends on collaboration both within our department and with colleagues elsewhere. Our computer systems are deliberately set up to *encourage* collaboration among their users, and that means encouraging easy access to one another's systems. This policy requires some degree of trust among the participants. If you've ever taken anything out of a safe deposit box at a bank, you know that it's possible to design a high-security shared facility, but that the cost is making it a big pain in the neck to use the secured data. Some computer systems are designed to have bank-level security, and everyone will think you're very clever if you figure out how to mess up such a system. Nobody will think you're clever if you mess up the 61A system.

The form you sign when you get your computer account says that it is for your use only and for course work only. We are not unreasonably strict in enforcing this rule. Nobody minds if you occasionally play a computer game late at night if it's the kind that doesn't wreck the keyboards or mice through repeated high-speed banging on one button. Nobody will object even if you occasionally bring a friend to play the game with you or if you write an occasional English paper on this facility instead of the official English Department computers. But if you are asked to give up the terminal by someone who wants to do course work and refuse, that's unacceptable. Remember, you and your fellow students are the ones who suffer from such obnoxiousness; the faculty and staff have other computers to work on.

In addition, you should know that, on occasion, our file servers go on the blink. You can detect this situation by noticing that your terminal has suddenly stopped typing characters or you get a message along the lines of "`NFS server not responding...`". If this happens to you (and it will at least once!), don't panic; usually the server is back within minutes or hours with your data intact. Please do not put yourself in a situation where a couple-hour server crash will prevent you from completing your project on-time. "How

can I avoid such a horrible situation?" you may ask. By starting (and finishing) your assignments early, of course!

9 Network Etiquette

Our computer facility is part of a worldwide network that lets you communicate with other users both by electronic mail and by immediate connection if you're both logged on at the same time. You may find that the Internet, much like amateur radio, is a good way to make friends.

However, please remember that the network is *not* exactly like amateur radio, in that most of the people on our network are trying to get work done and don't want to spend time talking with you. Therefore, please do not send mail or `talk` requests to people whom you don't know. For example, if your best friend from home went to college somewhere else and you don't know his or her e-mail address, do not ask randomly chosen people at that college to locate your friend for you. (You can send mail to `postmaster` at most sites.)

The best way to get to know people on the net is to join newsgroups. The same program that you use for the class newsgroup will also let you subscribe to groups on an enormous range of topics, both technical and recreational. Most participants in these groups will welcome individual communication that's relevant to the newsgroup topic.

Here are a few rules of newsgroup etiquette: (1) Do not post to a group until you've read it for a couple of weeks, so you'll know what people consider appropriate topics for that group. (2) Do not post messages in which you quote all of someone else's long message and then add "Me too!" at the bottom. (3) Don't be sarcastic. If you're angry, wait until tomorrow to post your message. Remember, too, that the other person isn't necessarily just like you; he or she may be eight years old, or eighty. (4) **Do not** post, mail, or forward chain letters! You will certainly lose your Berkeley computer account and may find yourself under arrest for fraud.

It is strongly encouraged that you subscribe to the group `news.announce.newusers` for more information about posting to newsgroups.

10 Readings, Homeworks, and Programming Assignments

You should complete the *reading* assignment for each topic **before** the lecture. For example, you should read K/R Chapters 1 though 4 by Tuesday. (Read COD Chapter 1 and sections 4.1 and 4.2 as soon as possible this week!)

Every week there will be problems assigned for you to work on. These assignments come in four forms:

- **Pre-lecture Quizzes** are short assignments designed to verify that you have read the assigned reading for the day's lecture. These quizzes are mandatory, are due at the beginning of the lecture for which they are assigned, and may be submitted on-line at the course website. The first pre-lecture quiz is for tomorrow's lecture, and is available now.
- **Laboratory exercises** are short, relatively simple exercises designed to introduce a new topic. Most weeks you'll do these during the scheduled lab meeting following Tuesday and Thursday's lecture. You will receive credit for these exercises by being *checked off* by your TA, normally during laboratory hours.
- **Homework assignments** consist mostly of more difficult problems designed to solidify your understanding of the course material; you'll do these whenever you can schedule time, either in the lab or at home. You may be accustomed to homeworks with huge numbers of boring, repetitive exercises. You won't find that in here! Each assigned exercise teaches an important point.

  Generally, there will be two homework assignments per week, but both are due on the Sunday after they are assigned. These assignments are included in the course reader. (The first assignment is also attached to this handout.) You are encouraged to *discuss* the homework with other students. Specific Homework requirements and grading policies are below.

- **Projects** are larger assignments intended both to teach you the skill of developing a large program and to assess your understanding of the course material. There are four projects during the term, and you might have you work on some of them in groups. Specific Programming project requirements and grading policies are below.

Everything you turn in for grading must show your name(s), your computer account login(s), and your working group number for group assignments. Please cooperate about this; make sure they're visible on the *top* of the files you turn in, not buried somewhere in a comment or a function.

11 Testing and Grading

The grading policy of the course has these goals: it should provide a reasonably accurate measure of your understanding of the material; it should minimize competitiveness and grade pressure, so that you can focus instead on the intellectual content of the course; and it should minimize the time I spend arguing with you about your grades. To meet these goals, your course grade is computed using a point system with a total of 300 points:

```
20 reading quizzes  @   0.5 points each  =  10 pts
15 labs             @     2 points each  =  30 pts
15 homeworks        @     4 points each  =  60 pts
 4 projects         @  12.5 points each  =  50 pts
 3 midterms         @    30 points each  =  90 pts
 1 final                                 =  60 pts
--                                         ---
58 assignments                             300 pts
```

There will be three midterms (set for the end of the third, fifth, and seventh weeks of the term) and a final. The exams **MAY** be open book, open notes – I will let you know. I want to remove time pressure from the list of factors affecting your midterm scores, so we'll hold the midterms on Fridays 'round Noon, (Room TBA), instead of during the lecture hour. My goal will be to write one-hour tests, but you'll have at least two hours to work on them. The relatively large number of midterms is meant to help you learn to take tests, and to reduce your anxiety about ruining your grade by having a bad day. Each midterm is worth only ten percent of your final course grade, so it is not the end of the world if you do poorly on one.

Each letter grade corresponds to a range of point scores: 280 points and up is an A+, 270–279 is A, and so on by steps of ten points to 170–179 points for a D−.

```
A+  280-300     A  270-279     A-  260-269
B+  250-259     B  240-249     B-  230-239
C+  220-229     C  210-219     C-  200-209
D+  190-199     D  180-189     D-  170-179
```

This grading formula implies that **there is no curve**; your grade will depend only on how well you (and, to a small extent, your partners) do, and not on how well everyone else does. (This grading scale may be a little bit too tough. If everyone does exceptionally poorly on some exam or overall grades are not simply where I want them to be, I may decide the exam or the grading standard is at fault, in which case I'll adjust the grade cutoffs as I deem appropriate. But I won't adjust in the other direction; if everyone gets an A, that's great.)

If you believe we have misgraded an exam or assignment, **please wait until grading standards are published** before complaining to your TA. If, after reviewing the grading standards, you still believe a mistake has been made, please return the exam to your TA with a **typewritten** note explaining your complaint. The TA will carefully regrade *the entire assignment*, so be sure that your score will really improve through this regrading! By University policy, final exams may *not* be regraded; to make up for this, we will grade every final exam twice. Final exams may be viewed int he CS main office after final grades have been submitted.

Incomplete grades will be granted only for dire medical or personal emergencies that cause you to miss the final, and only if your work up to that point has been satisfactory.

12 Homework and Project Policies and Grading

In contrast to prior semesters, homework in this course will be done independently. You and your friends are encouraged to discuss the problems among yourselves, but the work that you turn in must be written and tested by you alone. Both of each week's homework assignments are due at 8:00 PM on the following Sunday. Both **homework sets must be submitted electronically** unless otherwise noted.

The purpose of the homework is for you to learn the course, not to prove that you already know it. Therefore, although the weekly homeworks will graded on correctness, you will be afforded an opportunity to recover points by improving your understanding of the material. If you receive less than 90/100 credit on a particular homework, you can sign up for a **face-to-face grading session with your reader**. During this session, you and your reader will re-cover the material you didn't understand on the homework. If you show sufficient improvement, the reader may adjust your score. Sign-up sheets for the face-to-face sessions will be posted in the laboratory (and perhaps online). **Please bring a paper copy of your homework to the sessions!**

The four programming projects are graded on correctness and style. The latter two projects will probably include face-to-face grading with your reader. **The programming projects must be turned in online as well as in the homework box;** the deadline is usually 11:59 PM on the second Tuesday after it is assigned (i.e. you have two weeks for each project), but there will be some exceptions. You'll get further instructions about this when the time comes.

Online turnin: You must create a directory with the official assignment name, which will be something like `hw3` or `proj1`. Put in that directory all the files that you want to turn in. Then, while still in that directory, give the shell command `submit hw5` (or whatever the assignment name is). We'll give more details in the lab.

Paper turnin: There are boxes with slots labeled by course in room 283 Soda Hall. (Don't put them in my mailbox or on my office door!) What you turn in should include transcripts showing that you have tested your solution as appropriate.

13 Collaborative Learning Policies and Cheating

We encourage collaboration. It is the best way to learn and keep up with the wealth of material you are expected to cover. At the same time, cheating is not permitted. Sometimes the line between collaboration and cheating doesn't seem so easy to articulate, so we've tried to come up with very clear and enforceable rules so that you know what is expected and aren't uncomfortable collaborating, and, at the same time, so that those who break the rules can be held accountable.

Unlike the homework and projects, the tests in this course must be your own, individual work. I hope that you will work cooperatively with your friends *before* the test to help each other prepare by learning the ideas and skills in the course. But during the test you're on your own. The EECS Department Policy on Academic Dishonesty says, "Copying all or part of another person's work, or using reference materials not specifically allowed, are forms of cheating and will not be tolerated." The policy statement goes on to explain the penalties for cheating, which range from a zero grade for the test up to dismissal from the University, for a second offense.

For the programming projects, copying others' work, whether from your friend who took the course last semester or from other current students in other groups is cheating. If you don't know how to do something, it's better to leave it out than to copy someone else's work. If you do learn something from someone else, and understand it now, then cite it as theirs. But be prepared to back up that you understand it without them around. If you do not cite it, it is considered plagiarism, and is again, cheating.

It is highly unlikely that different people would arrive at the exact same solutions on their own. We do

have programs to test for code similarity – and these programs are smart enough to know when only the variable names have been changed. Don't cheat–you do a disservice to yourself, to those you copy from, and ultimately, to the whole course as time is taken away from preparing lectures and answering questions to deal with cheaters.

For the homework assignments, before you develop your solutions to the problems you are encouraged to discuss it with other students, in groups as large or small as you like. **When you turn in solutions, you must give credit to any other student(s) who contributed to your work.** This does not mean e.g. 16 of you should turn in precisely the same work. It means that you may talk about it, work it out, try it, and then each person writes it up on their own. Working on the homework in groups is both a good way to learn and a lot more fun! If you take the opportunity to discuss the homework with other students then you'll probably solve every problem correctly.

In my experience, most students who cheat do so because they fall behind gradually, and then panic at the last minute. Some students get into this situation because they are afraid of an unpleasant conversation with an instructor if they admit to not understanding something. I would much rather deal with your misunderstanding *early* than deal with its consequences later. Even if the problem is that you spent the weekend stoned out of your skull instead of doing your homework, please overcome your feelings of guilt and ask for help as soon as you need it.

If you are still unclear on the cheating policy, ask yourself this: in all of your talking with other students, did you UNDERSTAND the solution, or did you merely write down what someone else told you? If you didn't understand, then you aren't doing the work yourself. Again, it is better to have the answer wrong, or only partially right than to rely on someone else's answer. (Often because they too could be wrong!)

Penalties for cheating: Generally, the penalty for cheating on any assignment will be, at the very least, a zero on the assignment and will result in a notice being sent to the Office of Student Conduct. Further offenses and particularly egregious forms of cheating (like selling answers) will be dealt with more severely.

14 Lateness

A programming project that is not ready by the deadline may be turned in until 24 hours after the due date. These late projects will count for 2/3 of the earned score. No credit will be given for late homeworks, labs, pre-lecture quizzes, or for projects turned in after 24 hours. Please do not beg and plead for exceptions. If some personal crisis disrupts your schedule one week, don't waste your time and ours by trying to fake it; just be sure you do the next week's work on time.

By the way, if you wait until the night before to do the homework or a project, you will probably experience some or all of the following: a shortage of available workstations, an unusually slow computer response, or a file server crash.

15 Lost and Found

When people bring me found items from lecture or lab, I take them to the Computer Science office, 387 Soda. Another place to check for lost items is the campus police office in Sproul Hall.

16 Questions and Answers

Q: I am pre-enrolled for this course, and I'm planning to do the homework on my home computer. Do I still have to pick up a class account and log in by Wednesday to stay in the class?

A: Yes.

Q: I am a transfer student, and I'm pressed for time to fit in all my graduation requirements. I know how computers work. Do I really have to take 61C?

A: Yes, unless you have taken this same course elsewhere. However, your prior experience may well get you out of 61B, which is more nearly a standard second course. Mike Clancy is in charge of approving course equivalents.

Q: What's your advice on surviving this course?

A: Two things: Don't leave the homework and projects until the last minute, and **ask for help as soon as you don't understand something.**

Q: I got the Nobel prize last year, and my uncle is Chancellor of Berkeley. Do I still have to use my class account by Wednesday Noon to stay in the class?

A: Yes.

Q: I am disabled and need special facilities or arrangements to do the course work. What should I do about it?

A: If you need special arrangements about class attendance, taking tests, etc., I'll be glad to accommodate you; please take the initiative about letting me know what you need. For example, if you want to take tests separately, that's fine, as long as you ensure that we've worked out the arrangements before the test. The Disabled Students Program (ext. 642-0518) has voice response terminals from which blind students can connect to our computers. **If English is not your native language,** and you have trouble understanding the course materials or lectures for that reason, please ask for help about that too.

Q: I don't like (or have a conflict with) my pre-assigned discussion section. Can I switch?

A: You must negotiate this with the TA of the section you want to switch into. Please try to be settled into a definite section by the second week, when the group assignments will be made.

Q: I'm thinking about buying a personal computer. What do you recommend?

A: For this course, and in general for computer science courses at Berkeley, you don't *need* a computer of your own at all; you can work in the labs on campus. If you just want to be able to connect to the campus computers from home, anything with a modem will do. (If you live in certain dorms, there is an Ethernet connection in your room, and having a computer with an Ethernet adaptor will be very handy.) If you want to work entirely within your home computer, you can get the necessary software for PC-compatibles or the Macintosh in 387 Soda.

Some of our students, especially the ones with a particular interest in system administration, choose to run one of the free versions of Unix at home, usually Linux or FreeBSD, but to each their own. Learning to use some flavor of UNIX takes more effort than using commercial systems, but you learn a lot in the process.

Q: What should we call you?

A: "Kurt" is just fine.

Q: I'm having trouble understanding the assignments. I've never had a problem like this in school before. Does this mean I'm not as good a programmer as I thought, or should I just wait a week or two and see if things clear up?

A: Neither. THIS COURSE IS CHALLENGING! *There is nothing shameful about asking for help.* Every semester a few intelligent students end up in trouble in this course because they're too proud to come to office hours with questions. If you wait two weeks before you ask your question, by then you'll feel hopelessly behind, because the topics for those two weeks depend on the idea that you don't understand now.

17 First Assignments

Read chapter 1 and sections 4.1 and 4.2 of COD as soon as possible. Read chapters 1 through 4 of K/R by Tuesday and complete the reading quiz on the webpage. The first homework assignment is due next Sunday (check the reader or web site). You must log into your class account by Wednesday.