

Syllabus

ECS 156, Discrete-Event Simulation

Norm Matloff

Winter 2008

**PLEASE
FOLLOW
DIRECTIONS!**
Thank you very much.

Contents

1	What This Course Is About	4
1.1	Example	4
1.2	What Is the Programming Like?	5
1.3	Practical Value	5
1.4	Skills Gained	5
2	This Syllabus Is on Our Web Site	5
3	Consultation	5
3.1	Office and Office Hours	5
3.2	E-Mail	6
4	Prerequisites	6
5	Workload	6
6	Course Content (in Sequence)	6
7	Machines and Software	7
8	Course Materials	7
8.1	Printed Lecture Notes	7
8.2	Each Student Must Have His/Her Own Course Materials	7
8.3	Bring the Course Materials with You to Class	7
9	Tests	7
9.1	Terminology	7
9.2	Exam Schedule	8
9.3	Quizzes Schedule	8
9.4	Missed Tests	8
9.5	Open-Materials Policy	8
9.6	Work Only on the Test Sheet	8
9.7	How to Do Well on the Exams	9

9.7.1	Think, Not Memorize	9
9.7.2	Old Exams Yes, “Practice” Exams No	9
9.7.3	Homework-for-Learning Yes, Homework-as-Practice-for-Exams No	9
9.7.4	Concentrate During Class	9
9.8	An Embarrassing Subject	10
10	Bring the Materials to Lecture	10
11	The Discussion Section	11
11.1	Role	11
11.2	Quizzes	11
12	The Class Newsgroup and Web Site	11
13	Grading	12
13.1	Weighting	12
13.2	+ and - Grades	12
13.3	Exams Grade	12
13.4	Quizzes Grade	13
13.5	Homework Grade	13
13.5.1	Assignment of Letter Grades	13
13.5.2	You Must Turn in All Assignments to Pass the Course	13
13.5.3	You Should Get an A on the Homework	13
13.5.4	Back Up Your Files Frequently!	13
13.6	Accuracy of Records	13
14	Homework	14
14.1	Terminology	14
14.2	Announcements	14
14.3	Partners	14
14.4	Programming Style	14
14.5	Input Error Checking	15
14.6	Process for Submitting the Homework	15
14.6.1	Overview	15

14.6.2	Definition of “Files”	15
14.6.3	First Step: Put Your Files on CSIF	15
14.6.4	Package Your Files	15
14.6.5	Timestamp Your Package	15
14.7	Grading	16
14.8	If You Get Stuck	16
14.8.1	I Am Happy to Help If You Have Reached an Impasse	16
14.8.2	Use a Debugging Tool!	16
14.8.3	Help by E-Mail	17
14.8.4	Start Early!	17
14.9	Late Work	17
14.10	Reading Homework	17
15	I Do Care!	18

1 What This Course Is About

In discrete-event simulation (DES), we are interested in the evolution of a system in time, during which the changes to the system are abrupt (“discrete”) rather than gradual.

1.1 Example

As an example, let’s look at an assignment in the Winter 2006 offering of this course (details not shown here). The problem models the Kaiser health service’s advice phone line. One calls the number, waits in the queue, and is then served by an advice nurse.

The number of nurses available varies according to the number of people waiting. When a new call arrives, if the number of calls is greater than some threshold, a new nurse will be brought online. On the other hand, if there are fewer than a certain number of people waiting and being served for a specified time period, one nurse is brought offline.

Questions to be answered by the simulation:

- What is the long-run mean wait per call? Graph it against the values of the two thresholds and the length of the idle period.
- How much would all this cost, in terms of the long-run average number of nurses online? Again, graph the results against the parameter values.

1.2 What Is the Programming Like?

Some of you may have seen a bit of simulation in previous courses. DES has some similarity, but is a very different from a programming point of view. In discrete-event simulation, for instance, the issues of threads and data structures play a central role, while these rarely arise in ordinary simulation.

1.3 Practical Value

This is very practical material, used in industry, business and government. Many books have been written, and a number of commercial and open source software packages have been developed. (The first object-oriented language in history was SIMULA, a DES language.)

DES is considered a branch of computer science and it is often taught in CS departments. For example, the UCLA CS Dept. is a major DES research center. But it is also taught in departments of industrial engineering, e.g. at UCB, or in business schools, such as at MIT.

In the last offering of the course, a professor from the UCD School of Medicine was an active auditor. He had experience using simulation to develop management policies for hospital emergency rooms, and gave an excellent guest lecture. I hope to have someone else on campus to do the same this time.

1.4 Skills Gained

- discrete-event simulation programming, a skill used widely in industry, business and government
- the SimPy DES package (a set of APIs, accessed by Python)
- much of the Python language

It is NOT assumed that you have prior background in simulation, Python or SimPy.

2 This Syllabus Is on Our Web Site

Important note: This syllabus, and lots of other important information, is available at

<http://heather.cs.ucdavis.edu/~matloff/156.html>

3 Consultation

3.1 Office and Office Hours

My office is in 3053 Kemper Hall, Ext. 2-1953. My office hours will be Mondays, 2:00-3:00 and Tuesdays, 12:30-1:30. I will also be available after lecture for questions in most cases, though not when I have a meeting or seminar in the 3 p.m. time slot. I will also usually be available after discussion sections.

I enjoy my office hours very much, and look forward to interacting with you during them.

I am also available at other times if you have short questions.

3.2 E-Mail

You are welcome and encouraged to send me your questions via e-mail. I read my mail every day, including evenings and weekends.

4 Prerequisites

- Programming, at the level of ECS 60 (formerly course) 110. If you feel comfortable programming a linked list, then you're fine. **It is NOT assumed that you have prior background in Python or SimPy.**
- This is mainly a programming course, not a math course. Only basic probability concepts, at the level of STA 32 (or STA 131A, ECI 114, etc.) are needed, for specifying models and coding random number generators: Expected value, variance, density functions. This will be reviewed at the start of the course.
- Nonmajors are welcome!

5 Workload

There will be approximately four or five homework assignments. In the first assignment, you will do Monte Carlo simulation, which you will find is pretty straightforward. The remainder of the assignments involve discrete-event programming, which tends to require more involved thought processes. Most of you will find that the overall programming workload of our course will be perhaps 60-70% of what you saw in ECS 60/110.

6 Course Content (in Sequence)

- introductory examples of Monte Carlo simulation
- *world views* in discrete-event simulation programming (activity-scanning, event-list, process-oriented)
- introduction to SimPy discrete-event programming
- advanced SimPy
- statistical inference on simulation output
- random number generators, including variance-reduction techniques
- SimPy internals
- data structures for fast management of event lists

7 Machines and Software

Our base machines will be the Linux PCs in the CS instructional labs (CSIF). If you pre-enrolled in this course, or if you took a CS course last quarter, you should already have an account. Your account name will be your surname, and if that is not unique, it will be your surname followed by your first initial. Your initial password will be the last eight digits of your student number; change it immediately, using `yppasswd`. If you don't have an account, go to 147 Kemper Hall to obtain one. Note that all the instructional machines share a common file system, so you can access the files from any machine.

As mentioned above, you will be using Python and SimPy. These are available on the CSIF machines, but **ALL OF THESE ARE FREE, PUBLIC-DOMAIN PACKAGES**, available for all the major platforms—Linux, Microsoft Windows, Macs.

Thus you can work on your own machine at home. Note, though, that your programs will be graded on the CSIF machines, so make sure to try them there before you submit them. You will be submitting from the CSIF machines too. More on this in Section 14.6.

8 Course Materials

8.1 Printed Lecture Notes

As always with classes that I teach, there is no bound textbook. Instead, the text consists of my printed lecture notes (PLN), handed out in class. Also, they will be available as the PDF files in <http://heather.cs.ucdavis.edu/~matloff/156/PLN>.

8.2 Each Student Must Have His/Her Own Course Materials

Note that exams and quizzes are open-book, and thus there is **no sharing** of the PLN, books or other materials—including calculators¹—during exams and quizzes. **Thus every student must have his/her own copy of the PLN and other course materials.**

8.3 Bring the Course Materials with You to Class

Bring the PLN to all lectures, discussion sections and exams and quizzes.

9 Tests

9.1 Terminology

- the term *Exams* refers to examinations given in lecture; see Section 9.2
- the term *Quizzes* refers to examinations given in discussion section; see Section 11

¹Not that you need a calculator anyway; you don't.

- the term *Tests* refers to the Exams and Quizzes

9.2 Exam Schedule

There will be two Exams. Exam I will be given approximately at the fourth or fifth week of the quarter. Exam II will be held on the last day of instruction, Thursday March 13.

There will be no final exam. Exam II is not a final exam; it covers only material presented after Exam I, i.e. it is not cumulative.

9.3 Quizzes Schedule

See Section 11.2.

9.4 Missed Tests

There are no early or late Exams. Provisions for oral makeup Exams will be made if need be. Note: Oral exams are very difficult for the student, and are difficult for the professor to grade. Try to avoid this situation if possible.

Missed Quizzes are handled differently. See Section 13.4.

9.5 Open-Materials Policy

TESTS ARE TAKEN ON AN OPEN-MATERIALS BASIS.

It is mandatory that you bring the PLN with you to each Test. **Make absolutely sure that you remember to bring your PLN, as many of the Test questions will refer to specific pages in them.**

You are also welcome to bring whatever other materials you wish, e.g. your old computer science and statistics texts, dictionaries,² novels, comic books, whatever you want.

Note again that there is NO sharing of course materials, books, the PLN, calculators, papers, etc. during Tests.

9.6 Work Only on the Test Sheet

Please work ONLY on the test sheet, front and back. Do NOT turn in any supplementary sheets of paper. A deduction of 5 points will be made for not following this policy. Please, it's important.

²For example, do you know what the words *relinquish* and *discrepancy* mean? These words have shown up in problem statements in my Exams. We will be happy to explain any such words if you like during the Tests, but you may feel more comfortable with a dictionary.

9.7 How to Do Well on the Exams

9.7.1 Think, Not Memorize

The Exam questions test insight, not memorization.³ They assume *active* reading and *thinking* about the material.

If for example you encounter a statement in the reading which puzzles you, don't just skip it—ask me about it. An Exam question may later ask you to explain it.

9.7.2 Old Exams Yes, “Practice” Exams No

Please note carefully that I do not give practice exams.

This is because I believe that learning does not consist of memorizing, and because this course (in the way I teach it) is not like a “calculus-style” course.

Calculus exams have predictable, technique-oriented content. In a calculus class, you might have a lecture on integration by parts, then have a homework assignment of integration-by-parts problems, and then on the exam there would be a couple of such problems.

This would be impossible in the courses I teach, because every Exam problem is unique, and the problems test understanding of concepts, rather than techniques. Thus, it would be impossible to have practice exams.

I do make my old Exams available on the Web, but they are NOT practice exams. Instead, I make them available so you can get an idea of the general “flavor” of my exam philosophy.

9.7.3 Homework-for-Learning Yes, Homework-as-Practice-for-Exams No

Similar to the point that I do not give practice exams, it is important to understand that the homework is not considered practice for Exams either. The goal of the homework is to give you a chance to think more deeply about the material, and to give you a chance to nonpassively implement some of the course concepts. Sometimes an Exam problem will indeed be similar to some homework problem, but most Exam problems do not look like any homework problem.

Again, the best way to prepare for Exams is to constantly sift the material through your mind in a nonpassive manner, thinking about questions involving How?, Why/Why Not? What is the goal? and so on. The homework will really help you prepare for Exams in this sense, because it will make you think thoroughly about the course material, but you should not expect problems to be explicitly similar to the homework.

9.7.4 Concentrate During Class

MAKE SURE TO LISTEN CAREFULLY IN CLASS.

Weird statement! What does it mean?

³I've deliberately capitalized the word *Exam* here, in order to distinguish from Quizzes. Quiz problems are supposed to be very easy applications of what you've learned, while Exam problems probe more deeply.

Many Exam problems will come from the answers I give to questions students ask during lecture, or will be related to off-hand remarks I make during lecture when I suddenly get an inspiration.

Therefore, listen carefully to questions students ask in class, and to the answers I give. Make a note of seemingly-casual comments I make in lecture, and think about them carefully at home. These things often pop up in Exams. Such questions will still be answerable from the course materials if you miss class, but there is a definite advantage to paying close attention in class.

9.8 An Embarrassing Subject

An embarrassing subject which nevertheless must be mentioned is cheating on Tests. **If a student is found to be cheating, it will be treated as a VERY SERIOUS matter, not a harmless prank; it will harm his/her standing at the university, and also possibly make it difficult for him/her to get a job when seeking employment after graduation.**

Official university policy⁴ forbids

“Wandering eyes,” talking during exams...or leaving the exam room without permission.

Please work as follows during Tests:

- Try to sit in alternate seats if possible.
- **ABSOLUTELY NO TALKING** to classmates at any time during the Test.
- Keep your Test paper covered when you are not actually writing, so that it will not be so exposed to view.
- It would be greatly appreciated if you **USE THE RESTROOM BEFORE THE EXAM STARTS, RATHER THAN DURING THE EXAM.** And never leave the exam room without permission.

During a Test, if your paper is exposed, I will come to you and mention this to you. Thanks very much for your help.

There are ethical issues with homework too. It is OK if you discuss the homework with students not in your group, either in person or in the .d newsgroup. **However, sharing specific code is not allowed, and is considered plagiarism.**

10 Bring the Materials to Lecture

Make sure you bring the PLN to every lecture. They will play very active roles in the lectures.

I typically will ask you to open to a certain page, and then will call your attention to various sentences, examples and pictures on that page. The lectures will consist mainly of discussion and amplification of the examples in the PLN, with additional examples and comments. I will seldom write on the blackboard.

⁴See the *Class Schedule and Registration Guide* or the Web site of the Office of Student Judicial Affairs, <http://sja.ucdavis.edu/integ.htm>.

A MAJOR GOAL OF THE USE OF PRINTED LECTURE NOTES IS TO GIVE STUDENTS A CHANCE TO ASK QUESTIONS AND TO ENGAGE IN CLASS DISCUSSION. You are very much encouraged to bring up any questions you might have. Please don't be shy about asking questions; there is no such thing as a "dumb" question. Similarly, listen carefully to the questions the other students ask; this can be a very valuable source of insight, to which you should pay special attention.

11 The Discussion Section

11.1 Role

This quarter there are four courses that do not have a TA; ECS 156 is one of them. So, I will handle the discussion sections myself.

In most discussion sections, I will give a Quiz; assume there will be one each week, unless told otherwise. I will also use the discussion section time slots for interactive homework grading; this will be done in my office, on announced days.

11.2 Quizzes

A Quiz will be given almost every week in discussion section. Quizzes will not be announced; assume that there will be one each week.

Quizzes will be short (10 minutes) and simple. The latter means that **Quiz problems will be very straightforward and much easier than Exam questions.** Their purpose is to ensure that you are keeping up with the material.

The Quizzes typically will be one or two questions in length.

12 The Class Newsgroup and Web Site

The class newsgroup will be **ucd.class.ecs156**.

THE NEWSGROUP WILL PLAY A MAJOR ROLE IN OUR COURSE. IT IS REQUIRED THAT YOU READ THE NEWSGROUP EVERY DAY. Not only will announcements of homework assignments be posted there, but also there may be later corrections, hints, etc. And of course, dates of Exams and other related information will be announced on the newsgroup too.

You can read newsgroups using special-purpose programs such as **tin** or **slrn**; just type the program name, e.g. type `tin` when logged onto one of the CSIF machines. If you prefer GUI software, try **pan**. Or you can access newsgroup through general-purpose programs such as Web browsers.

Note that the newsgroup can be accessed only from a UCD Internet address. You will have this if you are using a campus computer or have connected to the Internet via the campus modem bank. If you are not in this setting, you can still read the newsgroup by using **ssh** to log on to one of the CSIF machines. and then typing `tin` or `slrn` at the Unix command line. Or, you can access the newsgroups directly if you go through the campus authentication service, at <http://email.ucdavis.edu/news/news-succeed.html>.

Please note that “.d” newsgroups are for discussion among students. This was the original intention when they were established some years ago; it seems that instructors of ECS 30 and 40 have abandoned that concept, but I feel it is important for the students to have a place of their own, without interference from the instructor.

Thus I usually do not read the .d groups. If there is a message you want me to see, please send me e-mail, instead of posting a message which you hope I will see on the .d group. (Do not post to the main newsgroup either.

13 Grading

13.1 Weighting

The formula used is

$$\text{course grade} = 0.6 \times \text{Exams grade} + 0.2 \times \text{Homework grade} + 0.2 \times \text{Quizzes grade}$$

where the Exams, Homework and Quizzes grades are each on the 4-point scale (4 for an A, 3 for a B, etc.).

13.2 + and - Grades

The threshold for a grade of n is (n-1).85; the threshold for an ‘n-’ grade is (n-1).6; the threshold for a ‘+’ grade is n.2. So, for example, if your weighted average from the above formula is between 2.6 and 2.84, your course grade is B-; if the average is between 2.85 and 3.19, your course grade is B; if the average is between 3.2 and 3.59, your grade is B+.

13.3 Exams Grade

Each Exam grade will be used in letter grade form only (the numerical grades are recorded but not used).

IMPORTANT NOTE: If your first Exam letter grade is lower than that of the second, other two Exams, and if your Homework and Quizzes grades are at least B, that first Exam grade will be thrown out completely. **This, coupled with the fact that most of you will get a A’s as your Homework and Quizzes grades, typically makes an important difference in the course grade for many students.**

By the way, though, please note that **this policy of course does not imply that you should not try very hard on the first Exam**, for many reasons:

- It might turn out that you do very well on the first Exam, better than one of the other two, so you definitely would want to have the benefit of that first high grade.
- When a student asks me for a recommendation to an employer, or asks me to write a letter for graduate school, I take ALL Exams into account.

Exam grading is noncompetitive (there is no “curve”), so it is possible for everyone to do well.

13.4 Quizzes Grade

You will get a letter grade on each Quiz.

In recognition of the fact that on (rare) occasions you will not be able to attend discussion section, your two lowest (letter) Quiz grades will be thrown out. Your other Quiz letter grades will be averaged to produce your overall Quizzes grade.

It is assumed/intended that, similar to the homework (see below), **most of you will get an A grade overall for your Quizzes.**

13.5 Homework Grade

13.5.1 Assignment of Letter Grades

Your letter grade on this homework will be as follows, based on your average homework score: 90-100%, A; 85-89%, A-; 80-84%, B; 70-79%, C; 60-69%, D; under 60%, F.

13.5.2 You Must Turn in All Assignments to Pass the Course

Note that all homework assignments must be completed in order to get a passing grade in the course.

In order to be considered “completed” the basic work must be done, though there may be errors. In the case of a programming assignment, for example, if the program has sections obviously missing, or if the program does not even compile, it is not considered complete.

13.5.3 You Should Get an A on the Homework

I aim for the vast majority of the class to get an A on the homework. Lots of help is available, so this aim should be achievable. (And it is; I use this policy in every course I teach, and it always works out that most people get A homework grades.)

The most important part of getting a good homework grade is to **start an assignment as soon as it is assigned.** Don't wait until a few days before the due date to start. **And make sure you have read the PLN thoroughly before starting.**

13.5.4 Back Up Your Files Frequently!

Accidental deletion or destruction of your homework files will generally not be grounds for extending the due date.

13.6 Accuracy of Records

At the end of the quarter, it will be your responsibility to make sure that my records are correct as to your homework scores.

14 Homework

14.1 Terminology

- the *due date* means 11:59 p.m. of the stated date
- a *homework group* consists of one to three people who work on and submit the homework together; given that I will be grading the homework myself, I would appreciate it if you would work in a group of at least two people, but it's still OK if you wish to work by yourself; more on this in Section 14.3

14.2 Announcements

Homework assignments will be announced in the class newsgroup. The homework files themselves will be on the course Web site, so look there when an assignment is announced.

Occasionally there will be news about a current homework assignment, such as clarifications, hints and so on. These will be announced in the class newsgroup.

14.3 Partners

Members of a homework group must work **TOGETHER**, instead of having the members divide up the work.

14.4 Programming Style

What about good programming “style”—good indenting, meaningful variable names, helpful comments, etc.?

You will not be graded on style, but it is crucial for communications purposes. For example, suppose you are having trouble getting your program to work and you e-mail it to me for help. **If I can't read it, I won't be able to help you.**

Note that this means that you write your program in good style **from the beginning**. Do NOT insert indentation, comments, etc. only when you are finished with the program and ready to turn it in, which obviously defeats the purpose of doing these things in the first place.

The same points apply to top-down programming. Do this not because it is required for grading (it isn't), but instead because it will make it easier for YOU, i.e. make it easier for you to write and debug your program, and easier for me to read if you ask me for help.

And for the same reason, create a Makefile, again not because you will be docked points if you don't have it but because it makes your work (and mine, if you seek help from me) easier.

By the way, speaking of programming style: You may have been taught that global variables are “evil,” to be avoided like The Plague. That is far too extreme an attitude, and the reasoning used doesn't make much sense. Moreover, in many cases one must use globals. See a discussion on this at <http://heather.cs.ucdavis.edu/~matloff/globals.html>.

14.5 Input Error Checking

In the last section, I stated that (a) I do not deduct points for poor programming style but (b) you should make sure you have good programming style anyway, because it will help you. By contrast, concerning input error checking: (a) I do not deduct points if your program doesn't do error checking, but (b) I would actually prefer that you don't do it.

Input error checking is extremely important. There are many examples of various disasters occurring in real life due to lack of it. But the homework assignments are difficult enough as it is, without adding this, so I'd prefer that you don't do error checking.

14.6 Process for Submitting the Homework

14.6.1 Overview

Since we use interactive grading (see below), you will not submit your homework in the traditional way. Nor will you use some "turn in" program.

Instead, you will send me an e-mail message of the nature to be described below, to certify that you completed the work before the deadline.

14.6.2 Definition of "Files"

When you have finished all of a given homework assignment, you will need to put ALL your files into one package. These means your **.py** files in the case of programs. In the case of written work, it means your **.tex** files, your **.jpg** files if any, AND your **.pdf** files.⁵

14.6.3 First Step: Put Your Files on CSIF

If you have been working on your home machine, upload your files to CSIF, using FTP.

14.6.4 Package Your Files

Use the Unix **tar** command to collect all your files into one **.tar** file. For example,

```
$ tar cf Hwk2.tar q1.py q2.tex q2.jpg q2.pdf
```

would create the file **Hwk2.tar** from the files **q1.py**, **q2.tex**, etc.

14.6.5 Timestamp Your Package

Then run the Unix **sum** command on the **.tar** file to produce a checksum, which is kind of a code number which is (for all practical purposes) unique to that file. E-mail that checksum (two numbers, actually) and

⁵It is not acceptable to simply turn in the latter.

your **.tar** file name (but not the file itself), to me. The date and time on your e-mail will verify that you sent the mail before the deadline. Then later during the grading, I will run **sum** again, and the fact that the checksum has not changed will show that you have not changed the file subsequent to sending the e-mail.

14.7 Grading

We will use interactive grading. I will announce which discussion section times will be used for the grading. The groups will assemble outside my office, and I will grade one group at a time, with about five minutes needed per group.

In the case of group size larger than one, all members of the group must be present during the grading time. If a member misses that time, the other group member will be graded, but the one who misses the time must reschedule a time in order to receive credit for the assignment.

During a group's time slot, the group will log in to their account to access their files. They will then explain their homework solution, and in the case of a program demonstrate that it works. I will ask each member of the group questions to confirm that he/she understood what they did. For example, such questions might ask why you wrote your program in a certain way, what the roles of the variables are, what numerical values the variables may have at times specified by me, why you did not choose certain alternative designs, and so on.

IMPORTANT NOTE: Each partner must be ready to answer questions about ALL the submitted work. This is because the two partners are supposed to work together on all the homework problems, which is very important. As a means of insuring this, **I will assign separate grades for each group member. Normally these grades will be identical, but if there is a substantial disparity in the levels of understanding the different group members have regarding the assignment, I will assign different scores to each member.**

14.8 If You Get Stuck

14.8.1 I Am Happy to Help If You Have Reached an Impasse

Please note that, **unlike ECS 30/40, homework assignments here will NOT lay out a detailed recipe, with tons of hints, telling you how to do the programming problems.**

The work is expected to be challenging, and is designed to be thought-provoking. This thought-provoking nature of the assignments is the vehicle by which you get to really understand the concepts. You are not necessarily expected to see right away how to do an assignment. Instead, you are expected to spend a considerable amount of time pondering the assignment, gradually seeing more and more, until you finally see how to do the whole thing. It is that thought process which will develop insight into the course material.

I will be quite happy to help you, including giving you hints individually—but only if, after giving a matter considerable thought, you still don't see what to do. Once you have reached the point where you cannot go any further, we very much encourage you to seek help from us. We want you to do well on the homework!

14.8.2 Use a Debugging Tool!

You will help yourself greatly—both in this course and in all courses in which you do programming—by making good use of a debugging tool. See the debugging section of my Python tutorial for details.

14.8.3 Help by E-Mail

Again, you are welcomed and **strongly encouraged** to send me e-mail, at the address matloff@cs, for help on homework assignments. But you can help me if you keep the following in mind:

- **Please do NOT send me e-mail in HTML format, or with lines longer than 80 characters.** If you do, I will of course still answer it, but my answer will not be as clear or as detailed. This is because when replying to e-mail, I like to quote parts of the original message, making the reply clearer. I cannot do this easily if your mail is in HTML or has long lines.
- Make sure you have clear indenting and commenting. (Again, for your own benefit, do NOT add these at the last minute; use them from the very beginning as you write your program.)
- Make sure you've made a good effort to track down your bug using a debugging tool.

14.8.4 Start Early!

Please make sure to start the homework soon after it is assigned. Don't wait until the last couple of days before the due date, because **when the due date approaches, I tend to be inundated with e-mail messages asking for help on the homework, and thus my answers at that time will necessarily be short and without details, since I won't have time to say more.**

14.9 Late Work

An assignment is late if it is submitted to me after the due date.

If—this should never happen, but just if—you are late, you will be assessed a 5% penalty the first late day, and 10%-per-day penalty after that in your grade for that assignment. (Since e-mail is available every day, each of the seven days of the week counts as one day.) The maximum total penalty is 50%.

Each project group will be allowed a total of 2 late days over the quarter, time which is not penalized. You can use this as being late 2 days with no penalty on one assignment, or as being late 1 day with no penalty in each of two assignments.

In order to use this option, you must submit your work during the time you are using as your grace period. If for example you are 3 days late on an assignment, you will not be allowed to use your grace period time on that assignment; it will simply be counted as 3 days late.

Don't squander your grace period days on the first assignment. The subsequent ones will almost certainly be more difficult, so save your grace time for then.

I will keep the appropriate records as to how many days of grace period you have used..

14.10 Reading Homework

The reading—in the PLN—is extremely important, and should be considered “homework.” As mentioned earlier, approximately 30-50% of the exam questions will come from the reading.

After each lecture, do the corresponding reading. **Do the reading PROMPTLY, that same day, rather than getting behind.** This is the only way that you can get any insight from the reading; doing the reading the night before an exam will not work. **If you still have not done the reading as of the night before the exam, you actually will be better off not doing it, and getting more sleep instead, as cramming overnight won't be beneficial.**

15 I Do Care!

I wish to emphasize that I care very much that you succeed in this course, and I look forward to getting to know all of you.