

CSc 252 Computer Organization — Final Version

Spring 2008, Tuesday-Thursday, 12:30-1:45 p.m.

Description: Basic machine organization; elementary hardware concepts; interrupts. Machine operations and instructions; assembly language concepts and programming.

Instructor:

Patrick T. Homer

Email: patrick

Office: Gould-Simpson 811, 621-7267

Home: see the printed syllabus. Please do not call after 9:30 p.m. This is a local call from Tucson.

<http://www.cs.arizona.edu/people/patrick>

Office Hours: see the class web page. Note: My office hours are subject to change during the semester. I will update the class web page and will email the listserv when changes are needed.

You are welcome to stop by my office, or call my office, other than during my office hours. If I have time, I will answer questions. If not, I will make an appointment with you for a later meeting (in-person or on the phone).

TA's:

Harini Jayaraman

Email: harini

Office: GS 710-E, 621-4089

Manish Swaminathan

Email: manish

Office: GS 710-B, 621-4089

Office Hours: see the class web page. The class web page will be updated and email will be sent to the listserv.

Class Web Page: <http://www.cs.arizona.edu/classes/cs252/spring08>

Prerequisite: CSc 127B or CSc 227 or equivalents from other institutions. If you do not meet the prerequisites, please talk to me immediately or you may be administratively dropped from the course.

Required text:

David A. Patterson, John L. Hennessy

Computer Organization and Design: The Hardware/Software Interface, 3rd edition

2005, Morgan Kaufmann Publishers, ISBN 1-55860-604-1

Note: there is a revised printing version of the 3rd edition that is also available. It contains a number of printing corrections, and has printed copies of some of the material

that is available on the CD.

Either the 3rd edition, or the revised edition, is okay for this class. There is a CD that accompanies both versions. You do need the information on the CD, so if you are buying a used edition, make sure it includes the CD!

Email: Email questions should be sent to all of us (patrick, harini, manish), not just one. That increases your chances of getting a quick response. Note that we will cc each other on responses to your email.

When sending email, **always** include a subject line. The subject line must have 252 at the start, and must be “reasonable”. Bad examples include “help”, “a question”, “prog 1 help”, etc. Good examples include:

252 - Help on mips add error

252 - Clarification on turnin time

252: Need help with hexadecimal conversions

Email is a **very** useful tool in helping you understand the course material. This is true for other courses as well. Having a good subject line helps immensely in filtering the important email (that related to class work) from less important email.

Email should contain information that is relevant. Including previous email’s on a topic can be useful, but can also be a waste of time. Include information that is relevant; remove that which is not. Note that this means you should not automatically include previous email in the message.

Provide your name in the message. Do not rely on us knowing who you are based only on your email address. When asking about a program, include your cs login.

Use full sentences and write as clear a message as possible. Email that is poorly written simply results in us sending a reply that asks for more information. As an example, tell us specifically what your program is doing wrong, where (what line) it fails on, what output is wrong, which test case(s) do not work, etc. Tell us what you already know about the problem as this saves us troubleshooting time and prevents us from telling you something you already know.

Email summary:

subject line contains 252 and a short version of the contents

include your name and your cs login

provide as much information as possible

include previous email only when useful in understanding your message

Email that does not follow these guidelines will be returned unanswered.

Listserv: A listserv has been setup. I will add students to the listserv on January 18th based on the official class roster using your email.arizona.edu address. You can subscribe using a different address by sending an email message to listserv@listserv.arizona.edu with the body of the message containing the line:

subscribe comporg FirstName LastName

Note that all 4 parts have to be present. You will also have to (later) answer the email response from the listserv in order to complete the subscription.

You can remove an email address from the listserv by sending an unsubscribe message to remove an old email address. Help with an unsubscribe message or other listserv commands is available at: <http://listserv.arizona.edu/>. (And, we will answer questions about the listserv as well.)

The listserv is set up to always include 252 at the beginning of the subject line.

You can send email to the listserv using the address comporg@listserv.arizona.edu. Email that asks a question of general interest to the class is appropriate for the listserv.

Questions that are specific to your solution to a homework or program should be sent to our individual email addresses and not to the comporg listserv.

Grading: The final grade, subject to modification, will be determined by:

- 45% from the programming assignments (8 planned)
- 10% from the homework assignments (4 to 6)
- 35% from the quizzes, and
- 10% from the final exam.

The final grades may be curved, but a weighted total of 89.5 and above is guaranteed to be an 'A', 79.5% and above at least a 'B', 69.5% and above at least a 'C', and 59.5% and above at least a 'D'.

I do not post grades. I will periodically email your current grades to your CS department login. This email will show up in your email.arizona.edu mail. If you do not read email at that address, you will need to set a **.forward** to forward such email to your preferred email address. You can set forwarding using the Filters option from the webmail interface at <http://www.email.arizona.edu/>.

Quiz Policies: A 30-minute quiz will be given in-class on every other Thursday, beginning Thursday, January 31st (see specific dates below). Each quiz will be handed out promptly at 12:30 p.m. and collected at 1:00 p.m. A 5 minute break and a 40 minute lecture will follow each quiz.

- Late arrivals will not be granted extra time. Quizzes will be collected at 1:00 p.m.
- Missed quizzes may not be made up.
- Your lowest two quiz grades will be dropped.

Quiz Dates:

- Quiz 1: Thursday, January 31st.
- Quiz 2: Thursday, February 14th.
- Quiz 3: Thursday, February 28th.
- Quiz 4: Thursday, March 13th.
- Quiz 5: Thursday, April 3rd.
- Quiz 6: Thursday, April 17th.
- Quiz 7: Thursday, May 1st.

Re-grades: If grading seems incorrect, unfair, or inconsistent on a program, homework, or exam,

you may appeal for a re-grade. You may talk with the TA's or me about the re-grade. Re-grades on homeworks, programs, and quizzes will only be accepted during the seven days following their return. The re-grade does not necessarily have to be fully resolved, but the initial request must be made during the one week period. Re-grades on the final exam and the last homework/program will be accepted through Friday, September 26th — the end of the five-week period during which grade appeals can be initiated:

<http://catalog.arizona.edu/2007-08/policies/gradappeal.htm>

Do not question or challenge the decisions of the TA's on grades and on re-grades! The TA's can only assign grades, and change grades, according to instructions that I have given. If you do not agree with the TA's, talk to me.

Program Grading:

Assembler programs will be graded based primarily on whether the program does, or does not, work. In general, we will run your program against a set of test cases. For each test case, you will receive the points for the test case if your output is correct; you will receive zero points if your output is not correct. Note this is a binary function, you will get all or none of the points for each test case. For programs, there will also be points available for style (comments, indenting, readability), error-free assembly, and for satisfying specific requirements of the assignment (such as correct use of the stack, use of recursion when specified, etc.). In general, 60% to 80% of the grade will be based on the test cases. We will provide some test cases, along with the output of our solution prior to the due date of each assignment. We may use additional test cases during the grading process.

You will be able to request a re-grade on a program. Points lost due to factors other than test cases cannot be reclaimed by a re-grade. Points lost due to failing test cases can be partially regained, depending on the amount of change in your code from the original turnin. In general, the formula for this is: 90% for less than 2% changed, 70% for less than 10% changed, 50% for less than 25% changed. The amount of change will be based on the lines of code: comments, and blank lines will be not be counted. Variations to this formula may be applied to specific assignments; see the class web site.

To request a re-grade on a program, you will first need to fix your code (we are available to help with this!). Next, use the turnin command for the program to turnin your corrected code (turnin will be set to a different directory after the due date has passed, so a re-grade turnin will not overwrite your original turnin). Third, send us (patrick@cs, harini@cs, manesh@cs) an email requesting the re-grade. All three steps will need to be completed within the time limit specified for the assignment.

See the class web site for re-grade information specific to each assignment.

Important Dates:

Wednesday, January 16: First day to file for grade replacement option (GRO).

Tuesday, February 12: Last day to drop resulting in deletion of course enrollment and last day to file for GRO.

Tuesday, March 11: Last day for dropping class or changing to audit.

Thursday, May 15: Final Exam 11:00 a.m. - 1:00 p.m.

Computer accounts: You will need CS department computer accounts for this course. You can obtain these through the web page: <http://www.cs.arizona.edu/computing/services/> and use either the “Apply for a NEW account on the CS systems” or “Re-activate an existing but closed account on the CS systems” link. If you already have CS department accounts, you still need to do the apply process to update the department’s data base as to the courses you are taking. If you need physical access to the computer labs in Gould-Simpson, you can obtain a swipe card that will let you into the facilities; the CS computer labs are available 7/24.

Late work: Students are strongly encouraged to complete the programming and homework assignments on time. There are three reasons:

- You will not have time to work on an assignment after its due date is passed. It is more important that you proceed to the next assignment.
- Solutions to each programming assignment will be provided for your benefit after the re-grade date.
- Late assignments are burdensome, and detract from time otherwise available to help students with the current assignment.

Do not feel that your program or assignment must be complete to be turned in. Fair partial credit will be given.

However, exceptional circumstances do sometimes occur. For this reason, each student may use up to a total of three (3) late days for programming/homework assignments. A “late day” is defined as one minute up to 24 hours after the due date/time. Weekend and holidays count as late days. The three days may be used on any of the assignments, multiple late days may be used on one assignment. Once your late days are exhausted, work turned in late will receive a grade of zero.

Machine down-time will be taken into account in extending the due dates for programs only if persistent and repeated outages occur over the life-span of the assignment. You should therefore endeavor to complete each program well before the deadline.

Cooperation & Cheating: I encourage you to talk with your friends about the course work; both giving and receiving advice will help you learn. However, students are responsible for understanding and following the University’s Code of Academic Integrity. Briefly, you will not accept solutions from other persons, you will not give solutions to others, and you will not tamper with graded papers, code, or exams. It is OK to talk with other students about algorithms and general approaches, but each student must develop and write her/his own code, and each person must turn in her or his own work. I also consider misuse of your computer account a violation of academic integrity. The one exception to this is for test cases. You are allowed to share test cases and expected results of these test cases for the programming assignments. Students who violate the Code should expect at a minimum to fail the course and have charges recorded by the Dean of Students office. They may also be subject to other penalties. Refer to the “Code of Academic Integrity”

<http://dos.web.arizona.edu/uapolicies/>

for further details on what constitutes cheating, the penalties that may result, and the

procedures involved.

Exams: The dates/times for the quizzes and the final exam are firm. Each student must be careful to present a self-contained effort during each exam, and to keep her/his exam paper from easy display. Violations will be dealt with severely, with penalties that may exceed the value of the exam.

Attendance: Attendance is not recorded. However, each student is fully responsible for all material covered by reading assignments, lectures, and handouts.

Variations: Exceptional circumstances do sometimes arise. If such occurs in your case, please get in touch with me (not the TA's) as soon as practicable. This can be done in person, in writing, by phone, and/or by email.

Missed quizzes cannot be made up. However, if there is an exceptional circumstance that causes you to miss a quiz, please let me (patrick@cs) know. If you miss more than two quizzes when exceptional circumstances are involved, I can adjust the formula used to determine your grade for the course.