Syllabus
EE/CS 458
Computer Architecture
Fall, 2010
LEC 001 TR 12:30 - 1:45 EMS E237

1 Instructor[s]

<table>
<thead>
<tr>
<th>Lecturer</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jason Bacon</td>
<td><a href="mailto:bacon@uwm.edu">bacon@uwm.edu</a></td>
</tr>
<tr>
<td>EMS 942, 229-4557</td>
<td></td>
</tr>
<tr>
<td>Office: TR 10:00 - 11:00, 3:00 - ??</td>
<td></td>
</tr>
</tbody>
</table>

You may also contact instructors via email to schedule an appointment outside regular office hours.

2 Let Success Happen

In order to succeed at anything, there are a few essential steps:

1. Allocate enough time to do it well, so that you’re not always in a hurry.
2. Utilize that time wisely.
3. Relax and have fun with it. Enjoy the challenge of the moment and don’t obsess about the outcome.

Most failures in life are the result of self-sabotage. People often set themselves up for failure (e.g. by procrastinating or biting off more than they can chew) and then suffer the pain of trying to do the impossible.

Many people are afraid to set themselves up for success, because doing so leaves them with no excuse to fail. However, failure is nothing to get upset about. The truth is, if you’re afraid to fail, you’re ego is too big. Learn to welcome failures; they are inevitable, and nothing prepares you better for ultimate success.

All that really matters is how you respond to failure: Do you learn from it and adjust so that you’ll do better next time, or stubbornly refuse to adapt and remain stuck behind the same hurdle?

Once you learn to enjoy challenges and detach yourself from the outcome, and set yourself up for success without fear of failure, success will come naturally and seemingly effortlessly.

3 Balance your Life

Hard work, friends and family, and time alone should all be priorities in your life.

Most people lose sight of this and end up spending a disproportionate part of their lives chasing after money or achievements as if they’re the ultimate goal. Success isn’t worth much if you’re not happy.
Balance doesn’t come easy. It takes time to build the character and skills to accept responsibility, know our limits, cope with difficult people, and say no to excess.

4 Understanding comes from Doing

"I hear and I forget. I see and I remember. I do and I understand.”

-Chinese proverb

Very little learning occurs even in the best of classrooms. Teachers can only point the way, especially in college where so little time is spent in class. Besides, you cannot gain understanding by watching someone else. What you see and hear in class will not make you understand. You need to put it into practice on your own. Watching your teacher write code won’t make you a good programmer any more than watching Dan Jansen will make you a good speed skater. You have to invest long hours of practice, but that in itself is not enough. You also have to use that time wisely. Practicing bad habits will make it even harder to progress, so make sure study the procedures and techniques carefully before you practice them!

5 Understand your Brain

Learning new concepts and committing new knowledge to long-term memory involves growing new neurons and axons. This is a biological process that runs at its own slow, biological pace, and is driven by repetition. Hence, it is imperative that you start your work early and expose yourself to it repeatedly, so that your brain has time to re-wire itself in the process of learning. Don’t try to force learning: instead, give it the time and attention it requires and let it happen. Study and work on assignments a little every day, and your brain will absorb it much more easily.

6 Preparation and Thoroughness

If you follow these simple steps, in this order, you will do well in college:

1. Attend all classes.
2. Take thorough notes. (Write down EVERYTHING that’s discussed!)
3. Study your notes carefully before trying to apply the knowledge.
4. Read the appropriate material in your textbook. Read the book thoroughly, but as an overview; don’t try to absorb every detail. The details come from practice.
5. Do all of your homework individually.
6. If all of the above fail, ask for help well before deadlines.

If you have to miss a class (this should only happen if there is a true emergency), get a copy of the lecture notes from a classmate, review them, and discuss them with your instructor.

7 Be Self-sufficient

Be as independent as possible. If your success depends on the charity of others, you’re in a precarious situation. You also don’t want your coworkers to think of you as a burden.
That said, there is a time to ask for help, so work toward finding balance. If it's clear that you've made good choices in trying to solve a problem on your own, those you ask for help will respect you and be willing to help.

When you do ask for help, be attentive and take thorough notes so that you won't have to ask again.

8 Personal Conduct

Getting to know new instructors every semester, and figuring out how to best approach them, can be an intimidating process. Some are friendly and laid back, some are shy, some are pompous.

My belief is that all people should treat each other as equals, with humility and respect. This is what I try to do with my students, and I expect the same in return.

Respect does not mean feeding someone's ego or addressing them with prestigious titles. It simply means being honest and direct, and always giving the benefit of the doubt. If we disagree on something, be open about it (at the appropriate time and place) with the assumption that I will listen, consider your point of view, and be grateful for your honesty. You may not change my mind in the end, but you should not be afraid to try, and hopefully will not get frustrated if you fail.

Conducting open, honest discourse is one of the most important skills you can learn before entering the work force.

Note that this is my preference and policy. Obviously, other instructors may react differently to this approach, so always be aware of whom you’re speaking to.

9 Grading

Your official grade must be based primarily on measurements taken under controlled conditions (i.e. exams and quizzes). However, this is not necessarily the best measure of your progress. When assessing your own skills, it is more useful to look at your experiences doing the homework. In particular, are you completing and understanding the programming assignments independently? Don’t worry about speed at this stage. That comes with experience. All that really matters is that you eventually get it done, and understand what you did.

If I were king, I would abolish letter grades and simply report percentage and percentile (i.e. class rank) for each student. Unfortunately, letter grades are required, so I must create a grading scale. I do not use bell curves. They are statistically nonsense for a sample size of less than several hundred students, and they make each student’s grade dependent on the average ability level of the students in the class they happened to join.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90 - 100</td>
</tr>
<tr>
<td>A-</td>
<td>85 - 89</td>
</tr>
<tr>
<td>B+</td>
<td>82 - 84</td>
</tr>
<tr>
<td>B</td>
<td>78 - 81</td>
</tr>
<tr>
<td>B-</td>
<td>75 - 77</td>
</tr>
<tr>
<td>C+</td>
<td>72 - 74</td>
</tr>
<tr>
<td>C</td>
<td>68 - 71</td>
</tr>
<tr>
<td>C-</td>
<td>65 - 67</td>
</tr>
<tr>
<td>D+</td>
<td>62 - 64</td>
</tr>
<tr>
<td>D</td>
<td>58 - 61</td>
</tr>
<tr>
<td>D-</td>
<td>55 - 57</td>
</tr>
</tbody>
</table>

Homework is about 25% of the final grade, and exams/quizzes are about 75%.

There will be ten 10-point quizzes during the semester, and three 100-point exams, so plan for an exam or quiz in every week. Some weeks may require a double quiz to remain on schedule.

The total of all the quizzes counts as one exam. The lowest of the four scores will be replaced by the average of all four, provided that a sincere effort was apparent throughout the semester.

Exams and quizzes will cover everything discussed in lecture through the previous week (previous lecture in summer sessions), and will test in-depth knowledge of concepts covered in programming assignments. Final exams are not comprehensive in the traditional sense. They are not generally longer than the other two exams, and focus on material after exam 2. However, all CS courses are somewhat comprehensive by nature, so you will need to understand material from earlier in the
semester.

All programs will be graded on how they perform on weise.cs.uwm.edu or miller.cs.uwm.edu. I do not recommend doing programming on another computer. If you do programming on your own computer, it is your responsibility to retest the program on weise before turning it in. The process may prove trivial or quite difficult, depending on the tools you used to develop the code in the first place.

Grades are posted on D2L throughout the semester. Be sure to review them periodically and report any apparent discrepancies. Challenges to grades posted on D2L must be raised within two weeks after the individual grade item was posted.

10 Grading Policy

There are no makeup exams or quizzes, except in the case of a documented emergency. You will need a note from your doctor, employer, or other authority, with contact information, to verify the reason for your absence. In rare circumstances, a makeup exam may be given without documentation. The exam must be taken within 24 hours of the scheduled time, and the score will be reduced by 25%. This exception is only available for exams.

Late assignments are not accepted. You will be given ample time to complete assignments (assuming you have allocated sufficient time for the course), and should generally try to complete them well before the due date in case you run into trouble. Extensions are not given due to system outages unless the systems were down for a significant portion of the assignment period.

11 Computer Accounts

Students must use the class account on miller and weise for all programming assignments. Miller and weise are essentially identical with the following exceptions:

1. Passwords should be changed only on miller, since the password database on weise is synchronized from miller periodically.

2. You cannot read or send email on weise.

Use whichever system has the lighter load at the time. Sometimes one or the other is sluggish due to the number of students compiling and testing programs.

Privacy notes: CS department computers are public property. Instructors and other staff members can access your files on them. Do not store any private information on these computers. They are to be used only for academic purposes.

12 Communication

All students must check their PantherMAIL account and the email of the class account on miller on a regular basis. Your university email can be forwarded to another address through D2L. Your class account email can be forwarded by creating a .forward file in your home directory on miller, containing the email address to forward to. Your miller account will receive program feedback via email, so if you are concerned about privacy, do not forward it to an insecure email account such as Yahoo!, gmail, etc. You can read your email on miller using “pine”.

If you’re having trouble with anything in the class, don’t let it persist: If you can’t solve it on your own within a reasonable time, ask for help.
Program examples and other materials will be made available in /home/CS/cs458/pub.

13 Useful Tools

If you have a Unix system at home (this includes Mac OS X), you can use

```
ssh miller.cs.uwm.edu
```

from a terminal window to log into miller from your computer. (You can also use weise this way.)

For Windows users, I recommend PuTTY, a free terminal emulator that supports ssh connections. Windows users can also install the Cygwin tools (a collection of Unix tools compiled under Windows) and then use ssh as they would on a Unix system.

To download files from the pub directory, I recommend FileZilla, a free graphical file transfer program.

If you’re not familiar with PuTTY and FileZilla, just download them and try them out.

Do not use telnet, rsh, or rlogin, or ftp. They are not secure, since they all transmit your password over the Internet unencrypted.

14 Making the most of your time in school

Write a lot of code before you graduate. Students who do only what’s required for class assignments are demonstrating a lack of interest in their chosen field, and in education in general. Good employers will discard resumes that don’t show enough initiative to explore beyond course requirements.

There are several things that every computer science student should do before graduating:

- Do a long-term personal project of at least several thousand lines of code. The application and language used are a personal choice, and demonstrate to employers where your interests lie. This project will provide the experience of writing a sizeable program over a long period of time (which can’t be done as a class assignment), and will demonstrate the initiative to go beyond minimum requirements.

- Use programming to solve problems in other classes. There are many opportunities to apply programming in virtually any non-cs class you take. Be creative.

- Experiment with installing and configuring other operating systems. This will enhance your understanding of OS capabilities and develop highly valuable work skills. PC owners have a wide variety of FREE choices these days. There are dozens of Linux distributions, FreeBSD, NetBSD, OpenBSD, Dragonfly BSD, Open Solaris. Many can be installed and run under a virtual machine like Parallels, VMWare, VirtualBox, or QEMU. The Internet runs primarily on open source Unix-like systems such as FreeBSD and Linux. Learning these systems is a valuable and portable work skill in itself. There are jobs involving WEB development and systems management in virtually every city on the map.

- Build a computer from scratch. In today’s world, you can get most parts for free or almost free (on the curb, on the WEB, etc.).

15 Required Course Materials

16  Course Overview

The aim of CS 458 is to develop an understanding of computer hardware design. Computer architecture lies on the boundary between computer science and electrical engineering, and at the heart of computer engineering. This course is intended to serve CE, CS and EE students by exploring the hardware side of the CS/EE boundary in detail.

16.1 Prerequisites

Junior standing; CompSci 354(P); CompSci 315(P) or ElecEng 367(P). Consent required to audit.

16.2 Topics Covered (Order may vary)

1. Digital Logic Review
2. Register Transfer and Microoperations
3. Basic Computer Organization and Design
4. Programming the Basic Computer
5. Microprogrammed Control (Optional)
6. CPU Design
7. Pipeline and Vector Processing
8. Computer Arithmetic
9. Input-Output Organization
10. Memory Organization
11. Multiprocessors

16.3 Requirements for Graduate Students

The college requires that graduate students taking U/G courses do additional work beyond the basic course requirements.

30% of the course grade for graduate students will come from a project to design a new processor. The project will be due on the last day of classes.

Each graduate student should write a short proposal for their processor design project, and make an appointment to discuss it with the instructor as early in the semester as possible.

16.4 Exam Schedule

There will be an exam or quiz almost every week throughout the semester.

<table>
<thead>
<tr>
<th>Exam</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam1</td>
<td>Thu Oct 7 in lecture</td>
</tr>
<tr>
<td>Exam2</td>
<td>Tue Nov 9 in lecture</td>
</tr>
<tr>
<td>Exam3</td>
<td>Wed Dec 22 10:00 - 12:00</td>
</tr>
</tbody>
</table>
17 Academic Conduct

I teach with the assumption that every student is eager to learn and understand. I can’t think of any other reason to invest the time, effort and money it takes to complete any college course. Discussing subject matter with other students for the purpose of understanding is helpful, and encouraged. However, all homework assignments are to be done individually. If a friend asks for help, help them understand by explaining concepts and providing hints, but do not give away answers. No other student should ever see your class work. Doing so denies them the opportunity to learn, and cheats other students who have done their own work.

Cheating on exams or plagiarism are violations of the academic honor code and carry severe sanctions, including a failing grade and/or expulsion from the University. However, I would hope that fear of disciplinary action is not the major motivation for honest, hard work.

A detailed description of Student Academic Disciplinary Procedures may be found in Regents Policy Statements, UWS Chapter 14 and UWM Faculty Document #1686.

18 Students with Disabilities

If you need special accommodations in order to meet any of the requirements of this course, please contact the instructor as soon as possible.

19 Accommodation for Religious Observances

Students will be allowed to reschedule examinations or other work around religious observance, provided that the instructor is given reasonable advanced notice.

20 Military Duty

Reasonable accommodations will be made for students called for active military service during the semester, such that the student may complete the course work at an earlier or later date.

21 Incompletes

A notation of “incomplete” may be given in lieu of a final grade to student who has carried a subject successfully until the end of a semester but who because of illness or other unusual and substantiated cause beyond the student’s control, has been unable to take or complete the final examination or to complete some limited amount of term work.

22 Complaint Procedures

Students may direct complaints to the instructor or the chairman of the Computer Science Program. If the complaint regards an alleged violation a specific university policy, it may be directed to the chairman of the Computer Science Program or to the appropriate university office responsible for enforcing the policy.
23 Discriminatory Conduct

Discriminatory conduct (such as sexual harassment) will not be tolerated by the University. It poisons the work and learning environment of the University and threatens the careers, educational experience, and well-being of students, faculty, and staff.

24 Grade Appeal Procedures

A student may appeal a grade on the grounds that it is based on a capricious or arbitrary decision of the instructor. Such an appeal shall follow the procedures adopted by the College of Engineering and Applied Science (CEAS). These procedures are available in writing from the chairman of the Computer Science Program or the Dean of CEAS.

25 Additional Information

More on UWM policies can be found at http://www.uwm.edu/Dept/SecU/SyllabusLinks.pdf.