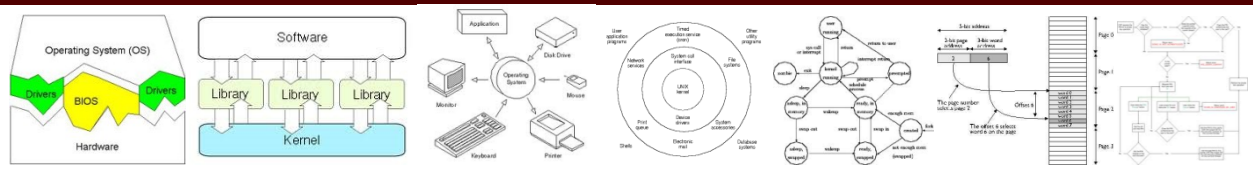
CSCE 313 – Introduction to Computer Systems

Department of Computer Science and Engineering

Syllabus
Textbook
Schedule
Grading
Labs
Lab Manual
Homework
Policies
Instructor Contact



CSCE 313 Introduction to Computer Systems
Course Syllabus
Fall 2010

Instructor: Ronnie Ward, PhD
Office: HRBB 338B
Office Hours: open door policy but email arranged appointments are encouraged
Office Phone: (979) 845-5534
email: ward@cse.tamu.edu

Time and Place: TR 9:35 PM - 10:50 AM; Room: 113 HRBB
Lab Section 501: M 01:50 pm-03:40 pm, HRBB 203
Lab Section 502: T 11:10 am-01:00 pm, HRBB 203

Teaching Assistant: TBD
Peer Teacher: TBD

Credit Hours: CSCE 313 is a four credit hour course

Course Description: Introduction to system support for application programs, both on single node and over network: OS application interface, inter-process communication, introduction to system and network programming, and simple computer security concepts; hands-on lab assignments.

Course Objectives: The objective of this course is to provide you with a general understanding of what system software is involved for an application program to run, both on a single node and over a network, and how this system software is to be used. In support of this, the course will prepare you to do system-level and network programming. This course will teach you how to "use" (as opposed to "design") system components, such as memory, file systems, process control, interprocess communication, and networking. By the end of this course you will have an understanding of the problems and pitfalls typically encountered in the design and implementation of multithreaded and networked applications and system

More specifically, by the end of this course you will be proficient at making full use of the services provided by the underlying operating system by programming directly at the operating system interface level, POSIX over UNIX in our case.

At the end of this course you will understand the following aspects of a computer system, in no particular order:

* Execution of a program; function calls; interrupts.
* Memory layout of a running program.
* What is an operating system; its components; why system calls; etc.
* The OS application interfaces; file system; memory control; process control; etc.
* Run-time environments; interaction of compilers, linkers, loaders to run a program.
* Concurrency, process synchronization, interprocess communication
* Network Programming; Berkeley sockets; RPC; pitfalls in networks.
* Security threats in centralized and distributed systems; authentication, authorization, confidentiality; security mechanisms.

Prerequisite: CSCE 312 or co-requisite CSCE 350

Required Text: UNIX Systems Programming: Communication, Concurrency and Threads, 2/E , by Kay and Steve Robbins, Prentice Hall, 2004.

Other Interesting Reading:

Computer Systems

- Computer Systems: A Programmer's Perspective, by Randal E. Bryant and David R. O'Hallaron

Operating Systems

- (hands-on) Operating Systems, Operating Systems, A Modern Perspective, by Gary Nutt (Addison Wesley)
- (not-so hands-on) Modern Operating Systems A.S. Tanenbaum (Prentice Hall)
- (similar) Operating System Concepts by Silberschatz, Galvin, Gagne, John Wiley and Sons, Inc., New York, 2004.

Systems Programming

- Advanced Programming in the UNIX Environment, by W. Richard Stevens (Addison Wesley)
- Advanced UNIX Programming, by Marc J. Rochkind, (Addison-Wesley Professional Computing Series)

OS Internals

- (Windows) Inside Microsoft Windows 2000, by D.A. Solomon and M.E. Russinovich (Microsoft Press).
- (Unix V) The Magic Garden Explained: The Internals of Unix System V Release 4 by B. Goodheart and J. Cox (Prentice Hall)
- (BSD Unix) The Design and Implementation of the 4.4 BSD Operating System, by McKusick, K. Bostic, M.J. Karels, J.S. Quarterman (Addison Wesley).
- (build-your-own; old, but excellent) “Operating System Design - The XINU Approach”, by D.E. Comer, 1984 (Prentice Hall).

General Programming

- "The Practice of Programming," by Brian W. Kernighan and Rob Pike (Addison-Wesley Pub Co; ISBN: 020161586X)

These optional books complement the textbook. Perusing them may help you better understand some issues discussed in class. A good selection will be made available at the Reference Desk in the Library.

Lectures: During lectures we will be covering OS concepts and case studies. The material covered in the lectures will provide the background and foundation for you to appreciate the lab assignments and to succeed in them. Reading of assigned portions in the textbook and other related material is your responsibility. You are also expected to follow instructions and be aware of announcements made during lectures. (See Communication Policy below.)

Labs: A very important part of this course is the Lab, where you will put into practice some of the material learned in the lectures, and where you will acquire a working knowledge of one widely used application interface (POSIX) to and operating system (UNIX). We will meet weekly for an in-lab session, where we will be presenting and discussing new material, or go over problems you may be encountering. After an introductory session to familiarize you with the particular environment that we will be using, there will be a series of machine problems with various difficulties, which will exercise different parts of the operating system: file system, process and thread management, memory management, networked execution, and so on.

In order to maximize the learning experience, we will not have groups of students work on the machine problems. Instead, each student will have to turn in their own solution. While we encourage you to collaborate during the Inquiry phase of the machine problem, the design and implementation of the machine problem solution has to be your own. You are not allowed to copy from other solutions of the machine problem, nor are you allowed to make solutions available to other students.

Machine problems are supposed to be handed in on CSNET. Details are available in the Lab Manual.

Note: Some of the lab assignments are quite demanding, and will require some dedication and some time. Expect that you won't be able to finish them during the allocated lab time!

Lab Grading scheme

Report	25%
Compilation	25%
Correctness and Completeness	25%
Design and Readability	25%

Components of Your Grade:

3 Exams (140+140+170)	450 points
4 Lab Projects	400 points
Homework	100 points
Other	50 points
Total points	1000 points

Grading Scale:

Percentage	**Grade**	**Comment**
901-1000	A	excellent work throughout the semester

801-900	B	above average work
701-800	C	average work
601-700	D	below average work
000-600	F	failure to do much of the work

Policies

Late-Submission Policy: All the submission deadlines have included extra time for the consideration of accidental events such as (not limited to) unavailable resource (machines are down,) sickness of students, stock market collapse, distress due to the Aggies losing a game, etc. This means that additional extensions are generally not granted. The rule of the game: START EARLY!

Both homework and projects will be submitted on CSNET. Unless stated otherwise, lateness is penalized with 1/5 of the earned points of the item per calendar day.

Examinations: There will be two midterm exams and one final examination. The midterms will be in-class, and the final will be during the allocated time during Final's Week.

All tests will be closed-book. You will be allowed one hand-written "cheat sheet" of size 8.5in x 5.5in. No other aids will be allowed, except for writing utensils.

Communication Policy: Instructor, Teaching Assistants, and Peer Teachers for this course will do their best to communicate relevant administrative information (deadlines, information about posted material, details about projects, locations of tutorials, and so on) in an effective and timely manner. We will be using announcements in class, postings on the web site, material on CSNET, and occasionally e-mails to students.

Having said that, keep in mind that this is not a distance education course! You are expected to be current with the material covered in class and with any announcements made in class. In fact, announcements in class will override whatever information has been made available through the other channels.

Note on e-mail accounts: Over the years we have been having problems with off-site e-mail accounts used by students. We will therefore be sending CSCE-313 related e-mail to the accounts listed on TAMUDirect only. (These are typically of the form xxxx@neo.tamu.edu, xxxx@cs.tamu.edu, or xxxx@tamu.edu.) Make sure that you have access to these e-mails, and forward them appropriately if needed.

Note on e-mail etiquette: E-mail is a very convenient and potentially effective way to communicate with instructor and TA, but only if used in a professional manner. Keep in mind that -- in particular when a deadline is looming -- we are receiving many e-mails, and all senders expect an immediate turn-around. Therefore, keep your e-mail short and to the point; indicate that you have done some thinking *before* typing the e-mail; provide necessary support documentation (e.g. code sections) when needed (don't attach huge amounts of code!); follow standard basic rules for courteous and professional communication; proofread your e-mail before sending it out. We will not answer e-mail that does not follow these rules. Repeated offenders will be added to the spam filter.

Absences: Lecture and lab attendance is expected. Attendance at exams is required. Infrequent unavoidable absences are understood, but each student is responsible for any missed material. For excused absences, students will not be penalized. See Section 7 of the Student Rules for the excused absence policy. Confirmation from your medical provider containing the date and time of the visit is required for all excused absences that are due to illness or injury. The Texas A&M University Explanatory Statement for Absence from Class form will not be accepted. You may not make up an exam unless your absence is official and excused. For unexcused absences, a grade of zero will be assigned for missed work.

Americans with Disabilities Act (ADA) Policy Statement
The Americans with Disabilities Act (ADA) is a federal antidiscrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you believe you have a disability requiring an accommodation, please contact the Department of Student Life, Services for Students with Disabilities in Cain Hall, Rm. B118, or call 845-1637.

Academic Integrity: "An Aggie does not lie, cheat, or steal or tolerate those who do."

Upon accepting admission to Texas A&M University, a student immediately assumes a commitment to uphold the Honor Code, to accept responsibility for learning, and to follow the philosophy and rules of the Honor System. Students will be required to state their commitment on examinations, research papers, and other academic work. Ignorance of the rules does not exclude any member of the TAMU community from the requirements or the processes of the Honor System

For additional information please visit: www.tamu.edu/aggiehonor

Student Rules: You are responsible for complying with all provisions of the Texas A&M University Student Rules.

Schedule: TAMU Academic Calendar Fall 2010 Final Exam Schedules

Week	Date	Topic	Reading	Homework	Labs
Week 1	Aug 30/31	Introduction; History; Architecture Support	Ch 1		
	Sep 1/2/3	OS Structures	Ch 1		
Week 2	Sep 6/7	Processes/Threads	Ch 2/3		MP1.1 out
	Sep 8/9/10	Processes/Threads	Ch 2/3		
Week 3	Sep 13/14	Posix Threads	Ch 12	HW1 out	
	Sep 15-17	CPU Scheduling			
Week 4	Sep 20/21	Critical Sections/Semaphores	Ch 14	HW1 due	MP1.1 due MP1.2 out
	Sep	Critical Sections/Semaphores	Ch 14		

	22-24				
Week 5	Sep 27/28	Thread Synchronization	Ch 13		
	Sep 29/30 Oct 1	Signals	Ch 8		
Week 6	Oct 4/5	Signals; Test Review	Ch 8	HW2 out	MP1.2 due MP1.3 out
	Oct 6/7	**Exam 1**			
Week 7	Oct 11/12	Timers	Ch 9	HW2 due	
	Oct 13-15	Memory, Paging, Demand Paging			
Week 8	Oct 18/19	Memory, Paging, Demand Paging			
	Oct 20-22	Unix I/O	Ch 15		MP1.3 due MP 2 out
Week 9	Oct 25/26	UNIX I/O	Ch 4		
	Oct 27-29	Files and Directories	Ch 5	HW3 out	
Week 10	Nov 1/2	Special Files	Ch 6		
	Nov 3-5	Posix IPC; Test Review	Ch 15	HW3 due	
Week 11	Nov 8/9	**Exam 2**			
	Nov 10-12	Posix IPC	Ch 15		
Week 12	Nov 15/16	Network Programming Intro	Ch 18		MP2 due MP3 out
	Nov 17-19	Networking	Ch 22		
Week 13	Nov 22-24	Sockets	Ch 22		
	Nov 25-26	Thanksgiving holiday			
Week 14	Nov 29/30	Transport		HW4 out	
	Dec 1-3	Transport			
Week 15	Dec 6-7	Security, Authentication, Crypto		HW4 due	MP 3 due
	Dec 8-9	Reading days, no classes			
	Dec 10	**FINAL for TR Sections**			

	Dec 15	**FINAL for MWF Sections**			

Handing in Homework: You are to submit your homework through the department's turn-in system. Proceed as follows:

- Log into http://csnet.cs.tamu.edu. (You may need to establish a VPN connection first)
- Follow the "My Courses" link.
- Select CSCE 313 from the given list of courses. You will see the class information and buttons for turn-in.
- Use the buttons to upload your files.
- NOTE: Subsequent submissions overwrite your earlier submissions for the same assignment!!
- If you have more than one file to submit, compress them into one file using `tar` or `zip`.

Notes on Handing in Labs and Homework:

1. Submit soft copy of everything that you should submit. The soft copies should be submitted using the CS department turn-in system before the deadline. For turn-in instruction see the course/lab webpage. The late penalty policy is as put in the course syllabus.
2. Compose the report in a Microsoft Word or Adobe PDF document and submit. Given the nature in addition to this word document you may have to submit text and C/C++ program files (with makefile). Without the makefile or compiling method described, any failure of compilation or running can cause points off. When submitting a lab assignment using turn-in system, pack all these files in a zip file and upload the zip file in the turn-in system. For turn-in system instructions see - turn-in.

CPSC 313 Lab Manual

Prepared by Dr. Riccardo Bettati, Zhongwei Jiang and Daniel Miller

In the CPSC 313 Lab we will be solving a number of increasingly more challenging system-programming machine problems. They will exercise our skills in making use of system resources and services, such as the file system, process control, process intercommunication, and networking and distributed computation. We will be using Solaris for this lab, which is a derivative of UNIX SVR4. In the following, you find a list of FAQ's for this lab, which should address most of the issues you may encounter when getting started with the lab. (If you should notice something missing in this document, please let us know, and we will address it.)

CPSC 313 Lab FAQ:

1. **Where can I find the course homepage?**
   The course home page for CPSC 313 can be found under http://faculty.cs.tamu.edu/ward/courses/313.f10/index.html . The homepage has all relevant information for the course, such as contact information of instructor and TA, links to slides and other information, and a link to the lab website. Follow this link to access information about lab assignments, resources, and other items.

2. **What is my user name and password for the Lab machines?**
   All students who register for courses in the CS Department are given a CS account. This account comes with a user name, and e-mail account and a home directory with considerable amount of disk space reserved. Your e-mail account will be of the type

   <xxx>@cs.tamu.edu, where <xxx> is your user id.

   Your user name and password to login to the Unix system will be the same as your email account.

   Details about claiming your CS account can be found at
   https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=19 .

3. **How do I turn in my labs?**
   Turn in your lab source code and other files using the submission system on CSNET. Details about this can be found at https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=24 .

   Each submission will typically consist of a Lab report, a directory with one or more source files, and a directory with one or more output files. Please follow the following structure and naming convention when submitting your Lab assignments:
   File and directory structure:

   - labreport.doc (or .pdf)
   - src (directory)
   - xxxx.c(or .cpp or .C)
   - xxxx.h (or .H)
   - ...
   - output (directory)
   - put any output files here

   Compress all the files and directories into one file with the name: <lastname>_<firstname>_<LabNo>.zip and then upload when you hand in your lab assignment.

   For example, if your name is John Roberts, and you are going to submit Lab 3, your compressed file name would be Roberts_John_Lab3.zip.

4. **Can I edit my files from my Windows account?**

   Yes; your home directory on your Windows account is the same as on your UNIX account. If you are more comfortable, you can edit the source files in a Windows environment, using any of the development environments such as Visual Studio, Borland C/C++ Builder, Eclipse and others. Any changes to the files will be visible from your UNIX account, where you compile and run your program.

5. **How do I compile and run my program on the UNIX machines?**
   The following link gets you to documentation that describes how to compile and run C and C++ programs in a

UNIX environment: https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=432 It covers the simple case of a single source file and that of multiple source files that need to be compiled and linked together.

6. **I want to complete my assignments from home! How can I access the CS VPN from off-campus?**
   The following link has documentation on how to access CS resources from either off-campus or over wireless through VPN: https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=186 It has documentation for various operating systems, such as Windows, Mac OS, Linux, and for dial-up users.

7. **How can I access the UNIX lab machines remotely?**
   If you work on a Mac OS or any other UNIX based machine, you can easily access the lab machines through command line tools, such as ssh or sftp. For Windows machines you can use tools like PuTTY or F-Secure. Documentation for ssh clients can be found at https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=158 and for file transfer clients at https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=9

8. **How do I use a UNIX system?**
   A description of basic UNIX commands, which should at least get you started, can be found at https://csnet.cs.tamu.edu/helpdesk/docs/view.php?doc_id=233 .

9. **UNIX has so many commands, functions, and system calls! Where can I find documentation?**
   The easiest way to access information about specific UNIX commands and calls is through the man pages. You access this documentation through the "man" command. At a shell prompt, simply type "man gcc" to know more about the gcc compiler command. The same counts for system calls or library functions. For example, type "man signal" to learn more about the "signal" function in the Standard C library. Type "man man" to learn more about man pages.

   There are many systems that support hypertext access to man pages. At the shell prompt, you can use the "info" command. (Type "info info" to learn more about it.)

   Alternatively, there are many resources on the Web, for example a hypertext version of the Linux man pages at http://linux.die.net/.

TEXAS A&M ENGINEERING
Department of Computer Science and Engineering
TAMU 3112
Texas A&M University, College Station, Texas 77843-3112