

# **CSCD 240 C & UNIX Syllabus Spring 2015**

## **CLASS INFORMATION**

<b>Class Meeting Time</b>	Monday through Friday 11:00am to 11:50am at CEB 228
<b>Instructor</b>	Yun Tian (Tony)
<b>Instructor Office</b>	CEB 304
<b>Instructor Office Hours</b>	Monday, Wednesday and Friday 9:30am to 10:50am Or by appointment
<b>Instructor Email</b>	ytian@ewu.edu
<b>Instructor Phone</b>	509.359.6162

## **COURSE DESCRIPTION**

This course includes program development tools of the UNIX operating system and syntax and programming techniques of the C language in that environment. UNIX topics include interactive shells, common text editors, utility programs, file system structure, libraries and operating system calls and system programming. C topics include data types, structures, pointers and pointer arithmetic, arrays, linked lists and function design and use. Programming projects are required.

## **COURSE OUTCOME**

After successfully complete this course,

- Students are able to utilize Unix/Linux shell commands to manage their files, including create, rename, copy and change permission etc.
- Students will understand and know how to use powerful shell utilities in Unix/Linux systems, such as wildcard, regular expression, grep and find.
- Student will learn and apply unique features of Unix/Linux systems, such as Piping and I/O redirection etc.
- Students will learn the concepts of processes and jobs in Unix/Linux system, and to manage processes and jobs using Unix commands.
- Students will learn and use tools and utility programs in Unix/Linux environment for C programming, including editors (vi or nano or emacs), compile driver (GCC), debugger tools (GDB) and deployment tool (Make), error check tool valgrind.
- Students will learn and apply in their assignments and projects various C topics, including C types, interactive and batch C programming, file I/O, function design, pointers and pointer to pointers (double pointers), structures etc.
- Students will understand and are able to apply what C differs from Java in low level memory access and management.
- Students will learn and be able to apply 2D dynamic arrays (represented as a double pointer) or linked lists in homework or project.
- Students will learn Unix/Linux system calls for low-level file IO.

## **COURSE WEBSITE**

EWU Canvas for CSCD240-01

All materials, notes and announcements are posted on Course Website. You should turn in your assignments and labs through EWU canvas.

## **REFERENCES**

### **C programming**

- Wikibooks, *C Programming*, [http://en.wikibooks.org/wiki/C\\_Programming](http://en.wikibooks.org/wiki/C_Programming)
- *The C programming Language* by Brian W. Kernighan and Dennis M. Ritchie. PDF is provided on the course website

### **LINUX/UNIX**

- Shotts, William, *The Linux Command Line*, PDF provided on the course website.

## **COMMUNICATION AND EMAIL**

During weekdays, I could reply your email within 24 hours since the time your email arrives at my mail box. If you send me an email on weekends, it will take longer for me to respond. Sometimes, if it is hard to answer your question through an email, I will simply request you to come to my office during my office hour. Please do not send me message through the Canvas inbox, I will not reply your message through Canvas online web interface.

## **ENVIRONMENT**

A true UNIX/LINUX environment is required for this class. (Debian 7 – 32/64 bit is required) running the gcc compiler. The lab computers are dual boot with LINUX already installed on them. In class, I will show you how to remotely log in cslinux.eastern.ewu.edu. To ease your **practice**, I will show you how to set up Debian 6 in Virtual Box.

## **TOPICS**

A tentative sequence of topics that we will cover this quarter

	Introduction to C and UNIX
	UNIX shells, file systems and basic file system commands
	Intermediate file system commands, pipes, grep, stdin, stdout, stderr
	Basics of C programming : C types, loops, boolean, bitwise operators, Editing, compiling, debugging programs
<b>Exam 1</b>	<b>Details will be announced</b>
	Debugging C programs Using GDB
	Variable Scope ( local and global variables)
	C arrays, pointers, pointer arithmetic, strings, malloc, calloc, free
	C double pointers, pointers as function parameters, return pointers
<b>Exam 2</b>	<b>Details will be announced</b>
	Files and file I/O, including high level and low level file I/O APIs
	Structures and typedef, linked list and multiple linked lists.
	Pointers as a member of structure type, pointers to structure variables
	Call by value and call by reference, Call by reference with pointers and double pointers.
	Make and makefiles, function pointers, void pointers.
	Linking, loading, advanced system programing.
<b>Final Exam</b>	<b>To be announced</b>

## **GRADING**

Type	% of Final Grade	Total
Exams	20%	2
Final Exam	30%	1
Homework	20%	
Labs	20%	
Quizzes	10%	
Attendance	0%	

95% - 100%	4.0
62% - 94%	Subtract 0.1
60% - 62%	0.7
0% - 59%	0.0

Some specific conversions: 95/4.0, 90/3.5, 85/3.0, 80/2.5, 75/2.0, 70/1.5, 65/1.0

The total weighted percentage point is calculated using this equation:  $\text{AverageExamsScore} * 20\% + \text{AverageAssignmentsScore} * 20\% + \text{AverageLabScore} * 20\% + \text{AverageQizzesScore} * 10\% + \text{FinalExamScore} * 30\%$

The instructor retains the right to increase grades above this formula.

**YOU MUST TURN IN ALL WORK, IN WORKING ORDER, ON TIME, TO PASS THE CLASS WITH A 2.5 OR BETTER**

## **GENERAL POLICIES**

**Americans with Disabilities Act:** Students requiring accommodations need to contact Disability Support Services (DSS) at (509) 359-6871. The DSS Office is located in TAW 125.

**Exams and Quizzes:** No makeup exams/quizzes will be given without prior arrangements having been made or in the case of a DOCUMENTED emergency after the fact. Exam dates will always be given at least several days before they occur. And review session or study guide for exam will be provided preceding each exam.

**Labs / Homework:** Labs and Homework assignments will be in the form of written and/or programming projects. They are due **BY THE REQUIRED TIME**. Thereafter, they WILL NOT be accepted. So please DO NOT ask for extension except documented or extreme cases.

Since the instructor will not post on website the solutions to labs or assignments, if you like to know the best answer to the lab or assignment problems, please **talk** to the instructor. The instructor can show you the solutions personally in office.

If you take this class, by default, you expect and agree that all assessments (including labs, assignments and exams) will be graded and published within **one week** (7 days) after the assessments are due.

**Professional Behavior:** You are expected to act in accordance with the department's [Code of Professional Conduct for Students](#). This means cheating will not be tolerated. If you ever receive aid on an assignment, be sure you document where the aid came from. Students who copy another's program/work **WILL** be subject to severe disciplinary action, which **WILL** include a course grade of 0.0 and a written statement in his/her permanent record.

**EWU Academic Policy/Cheating Policy:** You are allowed to discuss your thoughts with and help other classmates; however, you must do your own work, meaning ALL work should be your own. Plagiarism and cheating **WILL NOT BE TOLERATED**. Plagiarism includes using someone else's programs or parts of programs as your own; copying another person's work; handing in another person's work for your own. If you work with someone else to understand the content, it is important that each person does the assignment separately. Should any of your work be duplicated from another person or any other source, you will be asked to leave the class and not return. Your final grade will be 0.0. Any infractions will be handled in accordance with the academic integrity policy of Eastern Washington University and Washington State civil law. The best way for you to learn programming is to apply it to your own work.

**See Plagiarism/cheating. (WAC 132J-125-200)**

**Long version:**

- (1) Any student who, for the purpose of fulfilling or partially fulfilling any assignment or task required by the faculty as part of the student's program of instruction, shall commit plagiarism or otherwise knowingly tender any work product that the student falsely represents to the faculty as the student's work product, in whole or in part, shall be subject to discipline.
- (2) Any student, who knowingly aids or abets the accomplishment of cheating, as defined in subsection (1) of this section, shall also be subject to discipline.

**Short Version:** You do it - you fail.

It's not just a good idea, IT'S THE LAW!

**Incompletes:** Incompletes are subject to this restriction: "PASSING work/progress must be demonstrated (you must have a 2.5 or greater) through *three* weeks prior to the end of the term." Incompletes must be requested prior to finals week. Incompletes will not be granted without mitigating circumstances. (Dissatisfaction with your grade is NOT a mitigating circumstance!)

**Lectures:** Classroom activities will complement, not necessarily duplicate, the text. All posted lecture notes cannot substitute for your own notes taking in classroom. Furthermore, you are accountable for materials covered in class, so you **miss class at your own risk**. Your instructor will NOT teach the topics more than once in the lectures that a student skipped without acceptable excuses. Your instructor keeps checking class attendance. Examples given in class should be confirmed by the students to guarantee complete understanding of the subject.

**Cell Phone Policy:** You are expected to turn your cell phone/ringer off during class time. If you need to take a call please take the conversation outside of the classroom. If your cell phone goes off during class the instructor has the right to answer your phone.

**Rules About Seeking Help With Debugging:**

I **cannot** help you debug your homework, project or lab assignments. Generally, I cannot check line-by-line your homework/lab program to debug for you. Debugging code is one part of programming skill, and you need practice so that it can be improved. However, I could provide suggestions about how to debug and what steps you have to follow in order to pinpoint the bugs in your program.

**No Excuses Is Accepted As Listed Below: if you got any of these excuses, they are not acceptable.**

1. I do not know the deadline. I am not aware of the deadline.
2. I turned in a wrong version of my code. I turned in a wrong file.
3. I do not have a pdf creator, so I turned in a different format.
4. I do not know how to create a zip file that contains all the required source code and documents.
5. I did not double check my submission and I turned in a corrupted file.
6. My internet is down, there is no access to canvas. My Computer is broken.
7. My car is broken and I have to fix it.
8. I do not know how to ssh (remote login) into cslinux machine.
9. I do not know how to transfer files between my local computer and the cslinux machine, or to transfer files between the host machine and a virtual machine.
10. My anyConnect suddenly stops working and it works well before.
11. My virtual box or virtual machine stops working suddenly and they work well before.
12. I run into technical problems when I was turning in assignments. I keep getting errors from Canvas. Or I keep getting errors from the cslinux machine. ( All these technical problems have to be verified and confirmed by the IT department! )
13. I have **N** exams and **M** homeworks in my other **H** courses I am taking, please give me some extension.
14. I accidentally turned in my friend's homework as my own, while we were working together. This is considered as cheating and will be reported according to EWU academic policies.