



SYLLABUS: UNIX/LINUX SYSTEM ADMINISTRATION CITE 30103 - 045

COURSE & INSTRUCTOR INFORMATION

Course

Course Info: Unix/Linux System Administration, CITE 30103 – 045 (Class Number: 72034)

Semester and Year: Fall 20XX

Number of Credits: 3

Course Component Type: LEC

Course Webpage: TCU Online (<https://tcuonline.tcu.edu>)

Instructor

Preferred Method of Contact: Email

Response Time: Emails will be responded to within 24 hours.

Final Evaluative Exercise & Important Dates

Final Exam: December 9, 11AM-1:30PM
Midterm Exam(s): October 7, 12:30PM-1:50PM
Last Day to Drop: November 3
Last Day to Select P/NC: December 1
See TCU Calendar for detailed information: https://registrar.tcu.edu/fall-academic-dates.php

Note for students: The syllabus is your first course reading. It provides an orientation to, overview of the flow, and expectations of the course. You should turn to the syllabus for details on assignments and course policies.

Student Resources & Policy Information

Click or scan QR code for resources to support you as a TCU student. Please note section on [Student Access and Accommodation](#), [Academic Conduct & Course Materials Policies](#), and [Emergency Response & TCU Alert](#).



COURSE DESCRIPTION

Catalog Description

This course will cover the installation, management, and maintenance of Unix/Linux operating systems.

Prerequisites & Concurrent Enrollment

COSC 10403 or 10503; all with C- or better.

COURSE MATERIALS

Required Materials

- Textbook: [Linux+ and LPIC-1 Guide to Linux Certification 6th Edition](#) (Required)
Author: Jason W. Eckert; ISBN: 9798214000800
- Computer Hardware and Software needs: You will need to have a computer (e.g., PC or Mac) that is able to install a Virtualization Software (Desktop Hypervisor) (e.g., VMware Fusion for MacOS or Workstation for Windows and Linux), with enough computing power and disk space (minimum 50GB).
- [Programming in C](#) or any other Intro to C programming books as additional resources.

Supplementary Resources

Lecture notes/slides will be made available on TCU Online.

LEARNING OUTCOMES

Course Learning Outcomes

This course provides a hands-on introduction to Linux System, Git and C Programming. Upon successful completion, the students will be able to:

- 1) Explore, manage, and administer the Linux filesystems
- 2) Create, modify, and use basic shell scripts
- 3) Manage Linux processes
- 4) Create, modify, and delete user accounts
- 5) Install, configure, and use compression and backup software
- 6) Use version control with Git
- 7) Develop and run C programs in Linux
- 8) Apply control structures and functions in C to build modular programs
- 9) Work with arrays, strings, pointers in C
- 10) Implement structured data in C using structs, nested structs

COURSE REQUIREMENTS

Assignments

Homework

Homework will be based on lecture material and is designed to help you deepen your understanding of the course content. There will be a total of 12 homework assignments, accounting for 30% of the final grade. Students must have access to a laptop or desktop computer to complete all course assignments.

Quizzes and Class Participation

Students are strongly encouraged and expected to attend all lectures. All quizzes will be taken in class without exception. There will be a total of four graded quizzes, which will be pre-announced. Quizzes will constitute 10% of the final grade.

Exams

There will be two exams: A mid-term exam (30% of the final grade) and a final exam (30% of the final grade). Both exams will be held in class, without exception. Both exams will have components from lab and lectures.

Grading Philosophy & Policy

Late Work

Late homework assignments will be accepted up to 48 hours past the due date mentioned in TCU online. Late assignments will be penalized at 10% deduction per 12 hours. After 48 hours, you will get zero. Exceptions will only be given in extreme cases solely at instructor's discretion.

Participation, Engagement & Attendance

All students are strongly encouraged and expected to attend all lectures. While attendance may not be recorded daily, this course includes in-class quizzes and labs, and all exams will be conducted in class without exception.

Make-up Examination Policy

Make-up exams will not be provided by default, except in extreme circumstances. Exceptions may be made in cases of serious personal tragedy, severe illness, or similar situations, but such leniency is at the instructor's discretion. If unforeseen circumstances prevent you from attending class, please notify me so we can work together to ensure your success in learning the material.

Please note that I cannot accept medical documentation to justify absences, as accessing student medical records is considered an infringement on privacy. If you have a legitimate reason for your absence and want to provide verification, please access the [Absence Documentation Form](#) [here](#).

Course Assignments & Final Grade

Some grading item listed below may have opportunity for extra-credit of up to 0-10%.

Grade Item	Percentage
Homework	30%
Quizzes	10%+
Midterm Exam	30%+
Final Exam	30%+
Total	100%+

Grading Scale(s)

In the grading scale below, " [" means that the number is included in the range, and ") " means that the number is excluded.

Note: Each grading item will include bonus exercises as shown above indicated by "+", which allow students to earn extra credit. Apart from that, no additional extra credit or work will be offered under any circumstances. The final course grades will **NOT** be curved, with no exceptions.

Grade	Score
A	[94, 100]
A-	[90, 94)
B+	[87, 90)
B	[84, 87)
B-	[80, 84)
C+	[77, 80)
C	[74, 77)
C-	[70, 74)
D+	[67, 70)
D	[64, 67)
D-	[60, 64)
F	[0, 60)

Course Policies

Academic Honesty

I take academic honesty very seriously. Academic honesty is a core value of the university and all members of the university community are responsible for abiding by the tenets of the policy. You can find TCU's student's code of conduct (see section 3.4) in [student handbook](#) for

academic misconduct. Lack of knowledge of this policy is not an acceptable defense to any charge of academic misconduct. Examples of academic misconduct and course policy violation include but are not limited to plagiarism, cheating on examinations, unauthorized collaboration, falsification, multiple submissions and unauthorized public posting and distribution of instructor-prepared course material. If the occurrence of academic misconduct or course policy violation is proven, the student or students will receive an immediate and final grade of F. Disciplinary penalties will also be sought in addition to academic penalties. Names of the persons involved will be reported to the Dean of Students. This includes all parties involved, who will be treated equally, and I will not attempt to determine who actually developed the solution and who copied. For further clarification of university policies regarding academic honesty, please [read Academic Conduct & Course Materials Policies](#).

Technology Policy

Artificial Intelligence (AI) Ethical Considerations and Consequences for Misuse

Using AI-generated content inappropriately or without authorization may be considered academic misconduct and/or a breach of professional ethics specific to the discipline. Misuse of AI or similar assignment-help tools will be addressed in accordance with TCU's Academic Conduct Policy or other relevant policies, and could lead to penalties such as failing the course, dismissal from the program, suspension, or expulsion.

COURSE SCHEDULE

This calendar represents current course plans. Plans may need to change to enhance the learning opportunities and will be communicated via in-class announcements or TCU Online.

Week	Dates	Topics	Notes
Week 1	Aug 19	Introduction to Linux, shells, terminal basics, and commands (whoami, pwd, date, cal, man)	Read Syllabus HW1 assigned
Week 1	Aug 21	Navigating directories (ls, cd, pwd), viewing files (cat, less, head, tail)	
Week 2	Aug 26	Creating/removing files/dirs (mkdir, rmdir, touch), copying/moving (cp, mv, rm)	HW2 assigned
Week 2	Aug 28	Wildcards, hidden files, path shortcuts (., ..), file viewing options (ls -l, ls -a)	
Week 3	Sep 02	Permissions: ls -l, read/write/execute bits, chmod symbolic/numeric	HW3 assigned
Week 3	Sep 04	Default permissions, umask, ownership (chown concept), group project folder permissions	Quiz 1
Week 4	Sep 09	Searching with grep and find, viewing system info (uname, df, du, wc)	HW assigned
Week 4	Sep 11	Redirects (> , >> , <), pipes (), command chaining (;, &&,)	

Week 5	Sep 16	Environment variables (\$PATH, export), .bashrc basics	HW 5 assigned
Week 5	Sep 18	Command history (history, !), aliases, command substitution (\$())	
Week 6	Sep 23	Bash scripting basics: #!/bin/bash, running scripts, variables, echo, read	HW 6 assigned
Week 6	Sep 25	Arithmetic (\$()), if/else conditions, file tests (-f, -d)	Quiz 2
Week 7	Sep 30	Loops: for, while, until; case statements	HW 7 assigned
Week 7	Oct 02	Functions in Bash, script arguments (\$0, \$1, \$#, \$@)	
Week 8	Oct 07	Midterm (Linux topics)	
Week 8	Oct 09	Fall Break	
Week 9	Oct 14	Intro to C, program structure, compiling with gcc, basic data types, variables, constants	HW 8 assigned
Week 9	Oct 16	Operators, precedence, type casting, simple I/O (printf, scanf)	
Week 10	Oct 21	Control flow: if, if-else, switch	
Week 10	Oct 23	Loops: while, for, do-while, break, continue	HW 9 assigned
Week 11	Oct 28	Functions, parameters, return values, local vs. global variables	Quiz 3
Week 11	Oct 30	Recursion, header files, modular programming	HW 10 assigned
Week 12	Nov 04	Arrays: declaration, initialization, iteration, passing to functions	
Week 12	Nov 06	Strings: char arrays, null terminator, standard string functions	HW 11 assigned
Week 13	Nov 11	Pointers: *, &, pointer variables, arrays and pointers relationship	
Week 13	Nov 13	Structs: definition, member access	
Week 14	Nov 18	makefiles, Intro to Git	HW 12 assigned
Week 14	Nov 20	Git and Github essentials	Quiz 4
Week 15	Nov 25	Thanksgiving	
Week 15	Nov 27	Thanksgiving	
Week 16	Dec 02	Review & practice for C programming	
Week 16	Dec 04	Study Day	
Week 17	Dec 09	Final Exam (C topics)	