COMP 141 - Introduction to Computing Tools and Techniques

Course Syllabus – Fall 2024

Please note that this syllabus may be subject to changes within the first week of classes, mainly to enable minor corrections. Thank you for your understanding.

COMP 141 uses a common syllabus for all sections this semester. Please check LOCUS (and Sakai) to ensure you know your section and instructor's name.

Course Description

This course introduces the Unix shell environment and tools. Students who complete it will be fluent in the Unix shell as an instrument to operate computers optimally, helping them to succeed in future academic, research, and professional computing disciplines and needs.

Motivation

Despite the current era of rich graphical user interfaces (desktop, web, and mobile), the "command line" experience remains important, especially for software developers and computer-aided scientific researchers. Many development scenarios still require command line and fluency in Unix tools, including the modern embedded, cloud, cybersecurity, and supercomputing environments. With mobile computing and Internet of Things, Unix (notably via Linux) has risen to the level of a ubiquitous platform, owing to its lightweight and open-source nature. While this course does not aim at being a comprehensive programming class, students will master essential programming skills using shell scripting. They will also learn about computer-aided problem-solving using Unix commands supported by shell scripts.

Prerequisites

There are no prerequisites.

Course Delivery

The course is fully online, with asynchronous lecture videos and reading material, and weekly online synchronous meetings to let students ask questions and get content clarification from the instructor. We will use Sakai (https://sakai.luc.edu/) as the online platform to support, centralize, and store the course announcements, materials, activities, assignments, quizzes, progress reports, and grading communication.

Sections

Section	Sync Class Time	Instructor	Class Link	E-mail
001	Tu 4:15 PM – 5:30 PM	Daniel Moreira	https://luc.zoom.us/j/86020500730	dmoreira1@luc.edu
002	Mo 5:30 PM – 6:45 PM	John O'Sullivan	https://luc.zoom.us/j/83729984014	josullivan1@luc.edu
003	We 4:15 PM – 5:30 PM	George K. Thiruvathukal	https://luc.zoom.us/j/89470526329	gthiruvathukal@luc.edu
004	Th 4:15 PM – 5:30 PM	Satyaki Sikdar	https://luc.zoom.us/j/86247853182?pwd=zena6bs5czOvyQihZf3n1sEkCzCFJ0.1	ssikdar@luc.edu
005	Mo 4:15 PM – 5:30 PM	John O'Sullivan	https://luc.zoom.us/j/83257470524	josullivan1@luc.edu
006	We 5:30 PM – 6:45 PM	Corby Schmitz	COMP141-006	cschmit@luc.edu

Teaching Assistants

All the sections will share the following teaching assistants (TAs). Please feel free to contact them with questions about the course content and help with the assignments.

- Mr. Erik Pautsch (epautsch@luc.edu)
- Mr. Sagar Pyakurel (spyakurel@luc.edu)

Office Hours

In addition to the weekly online synchronous meetings, instructors will offer office hours in the following terms.

Section	Instructor	Modality	Appointment Link
001	Daniel Moreira	Online and in-person by appointment.	https://calendly.com/danielmoreira/fall-2024-office-hours
002	John O'Sullivan	Online by appointment.	https://calendly.com/josullivan1-h1c
003	George K. Thiruvathukal	Online and in-person by appointment.	https://calendly.com/gkthiruvathukal/students-only
004	Satyaki Sikdar	Online and in-person by appointment.	https://calendly.com/satyaki-sikdar
005	John O'Sullivan	Online by appointment.	https://calendly.com/josullivan1-h1c
006	Corby Schmitz	Online by appointment	Mo 5:30 - 7 PM Link Appointments: cschmit@luc.edu

Grading

Every activity matters. Every other week, there will be one practical **assignment**; short assignments will be worth 5 points each, while long assignments will be worth 10 points each. Every four weeks, an online multiple-choice **quiz** will be worth 10 points. The final exam will be an online multiple-choice quiz open during the finals week.

The final score will be the total points earned divided by the total points possible. This score will be converted to a concept grade according to the following scale.

Grade	Score	Grade	Score
A	>= 93%	C+	>= 77%
A-	>= 90%	C	>= 73%
B+	>= 87%	C-	>= 70%
B	>= 83%	D+	>= 67%
B-	>= 80%	D	>= 63%
		F	< 63%

Late Work and Resubmission Policy

Late work submission is accepted but subject to the following penalties.

Days Late	0	1-2	3-6	7-13	>= 14
Score Deduction	0%	5%	10%	20%	100%

Late penalties can be waived depending on the presentation of appropriate documentation justifying the delay.

Resubmission is allowed any number of times before the work deadline.

Makeup Policy

Activities such as quizzes missed without a pre-excused absence will be graded as zero. The only exception to this rule is a documented medical or health situation where the student could not provide prior notification.

The instructors reserve the right to give a makeup quiz completely different from the regular one taken by the other students. It may include arranging an alternative activity that requires quiz makeup outside regular class time with a short turnaround time.

Course Topics

The course content is heavily based on William Shotts' book "The Linux Command Line", freely available online at https://linuxcommand.org/tlcl.php. There are also additions made by the instructors. The covered topics are the following:

1. Preparing your computer – or a remote one – for operation: MacOS Terminal, Windows Subsystem for Linux, or Ubuntu Linux running natively.

2. Command-line operation: calling programs, and passing options and arguments.

3. Time-saving practices and notations: keyboard shortcuts and regular expressions.

4. Filesystem: files, directories, user and group permissions (chmod, chown), file and directory manipulation (cp, mv, rm, cat, mkdir, cd, ln, etc.).

5. Processes: running and monitoring programs, job, foreground, and background control, I/O redirection, stream commands (e.g., cat, grep, head, tail, tee).

6. Shell environment: reading and setting variables up, significant system variables (e.g., PATH, USER, HOME), and prompt customization.

7. Text editing: fast and advanced operation with Vi.

8. Program installation: package management, software installation, removal, and update.

9. Networking: remote access, file transferring, and examination of network configuration.

10. Archiving and file compression: usage of tar, zip, gzip, etc.

11. Text processing: usage of commands to quickly read and manipulate file content, such as cat, sort, uniq, cut, paste, join, diff, sed, etc.

12. Program compilation and interpretation: compiling programs in C and Java, and using interpreted languages such as Python.

13. Shell scripting: programming basics to demonstrate the potential of automating tasks in the shell.

14. Version control: Git profile creation, basic usage, and source code version control.

Tentative Schedule

Week	Topics	Assignment	Quiz
Week 1 (Aug 26)	Environment Setup, What is the Shell?	HW01	
Week 2 (Sep 2)	Navigation, Exploring the System		
Week 3 (Sep 9)	Manipulating Files and Directories, Working with Commands	HW02	
Week 4 (Sep 16)	Redirection, Seeing the World and the Shell Sees it		Q1
Week 5 (Sep 23)	Advanced Keyboard Tricks, Permissions	HW03	
Week 6 (Sep 30)	Processes, The Environment		
Week 7 (Oct 7)	Customizing the Prompt, Package Management, *Fall Break*	HW04	
Week 8 (Oct 14)	A Gentle Introduction to Vi		Q2
Week 9 (Oct 21)	Networking	HW05	
Week 10 (Oct 28)	Searching for Files, Archiving and Backup		
Week 11 (Nov 4)	Regular Expressions, Text Processing	HW06	
Week 12 (Nov 11)	Compiling Programs, Writing your First Script		Q3
Week 13 (Nov 18)	Keyboard Input with "read", Positional Parameters, Branching with IF	HW07	

Week 14 (Nov 25)	Looping with FOR, WHILE, and on Files, *Thanksgiving*		
Week 15 (Dec 2)	Version Control and Git		
Week 16 (Dec 9)	No classes, *Finals Week*		Q4

Learning Outcomes

- Acquiring skills required for success in high-end computing jobs.

- Mastering the Unix philosophy, which has lasting benefits when it comes to software development. By writing tools that are focused and do one job well, students will write software with fewer bugs that is more likely to stand the test of time. The Unix philosophy is related to pioneering ideas from other great thinkers, such as Mies van der Rohe, who advocated “less is more” as a fundamental design principle.

- Understanding that some traditions of computing are worth the investment of time and should be carried forward. Understanding Unix is helpful in understanding the remainder of the Computing disciplines. For instance, ideas such as regular expressions (first found in Unix) are wired into every modern programming language.

- Learning a text editor (Vi/Vim) remains important when working on remote servers, especially in supercomputing and embedded systems. Vi/Vim is not the only choice but is closely aligned with the university’s social justice system and has one of the most vibrant communities. The creator of Vim works to improve the conditions of refugees and children in Uganda!

- Being more confident in other introductory courses. Knowing how to edit and compile programs, write scripts, and work with Unix-like systems allows us to focus more on the ideas of computer science and programming fundamentals rather than fighting with the computer. Because every student will learn how to operate a computer in a more advanced and conscious way, the techniques learned in this course can and will be carried forward to all courses, even when students decide to use Windows (which has recently added support for Linux as a subsystem). The fact that Linux, MacOS, and Windows all offer Unix-like semantics means the command line is important to them and their business interests.

- Acquiring skills needed to succeed in systems, DevOps, and cybersecurity domains, where this course’s tools are crucial to problem solving.

- Preparation for the “hard” CS programming classes, e.g., programming languages, operating systems, distributed systems, high-performance computing, etc., which assume fluency with the command line.

Academic Honesty

Students are expected to read the statement on academic integrity, available at http://www.luc.edu/academics/catalog/undergrad/reg_academicintegrity.shtml. This policy applies to this course. The minimum penalty for academic dishonesty is a grade of F for that assignment. Multiple instances or a single severe instance on a major exam or assignment may result in a grade of F for the course. All cases of academic dishonesty will be reported to the department office and the relevant college office, which will place them in your school record.

Academic dishonesty includes, but is not limited to, working together on assignments that are not group assignments, copying or sharing assignments or exam information with other students except in group assignments, submitting as your own information from current or former students of this course, copying information from anywhere on the web and submitting it as your own work, and submitting anything as your own work which you have not personally created for this course. If you wish to use materials that are not your own, please check with the instructor beforehand and cite your source clearly. When in doubt, ask first! Be aware that the instructors may have changed the values for many textbook problems used for homework problems. For those problems that require open-ended answers, please be very careful to state the answers in your own words, not in the words of the instructor's manual, nor in the words of students who have previously taken this course.

Privacy Statement

Assuring privacy among faculty and students engaged in online and face-to-face instructional activities helps promote open and robust conversations. It mitigates concerns that comments made within the context of the class will be shared beyond the classroom. As such, recordings of instructional activities occurring in online or face-to-face classes may be used solely for internal class purposes by the faculty member and students registered for the course, and only during the period in which the course is offered. Students will be informed of such recordings by a statement in the syllabus for the course in which they will be recorded. Instructors who wish to make subsequent use of recordings that include student activity may do so only with informed written consent of the students involved or if all student activity is removed from the recording. Recordings including student activity initiated by the instructor may be retained by the instructor only for individual use.

Policy for Recording Live Sessions

In this class, software will be used to record live class discussions. As a student in this class, your participation in live class discussions will be recorded. These recordings will be made available only to students enrolled in the class, to assist those who cannot attend the live session or to serve as a resource for those who would like to review the content that was presented. All recordings will no longer be available to students in the class when the Sakai course is unpublished (i.e., shortly after the course ends, per the Sakai administrative schedule). Students

who prefer to participate via audio only will be allowed to disable their video camera so only audio will be captured. Please discuss this option with your instructor.

Religious Holidays

If you need to miss class for a religious holiday, please alert the instructor in advance so that it will not be counted as an absence. As required, due dates for quizzes and tests, homework and assignments can be adjusted if they conflict.

Students with Disabilities

Loyola University provides reasonable accommodations for students with disabilities. Any student requesting accommodations related to a disability or other condition is required to register with Student Accessibility Center (SAC), located in Sullivan Center, Suite 117. Professors receive the accommodation notification from SAC via Accommodate. Students are encouraged to meet with their professor individually in order to discuss their accommodations. All information will remain confidential. Please note that in this class, software may be used to record class lectures in order to provide equal access to students with disabilities. Students approved for this accommodation use recordings for their personal study only and recordings may not be shared with other people or used in any way against the faculty member, other lecturers, or students whose classroom comments are recorded as part of the class activity. Recordings are deleted at the end of the semester. For more information about registering with SAC or questions about accommodations, please contact SAC at 773-508-3700 or SAC@luc.edu.

Continuous Improvement

We believe in a personal quality process of continuous improvement. Anything can be improved by applying the quality process of “Plan, Do, Check, Act” (PDCA). To improve the course and the learning of programming and computer science concepts, we welcome your feedback, comments, suggestions, and complaints at any time.

We may ask you to participate in surveys and discussions during the course to support the PDCA process above. These surveys will measure student impressions of the course; when time permits, we will share the results with the class. Your input on these surveys is anonymous and does not affect your grade.

We will also be using a LUC feedback system called IDEA toward the end of the course to gather your feedback on both the course itself and the instructor’s teaching; we use this kind of feedback to improve how we teach the information in the course, and the CS Department uses it to help refresh and enhance our course offerings.