

Spring 2024

## CS 288-006: Intensive Programming in Linux

Mohit Dale

Follow this and additional works at: <https://digitalcommons.njit.edu/cs-syllabi>

---

### Recommended Citation

Dale, Mohit, "CS 288-006: Intensive Programming in Linux" (2024). *Computer Science Syllabi*. 414.  
<https://digitalcommons.njit.edu/cs-syllabi/414>

This Syllabus is brought to you for free and open access by the NJIT Syllabi at Digital Commons @ NJIT. It has been accepted for inclusion in Computer Science Syllabi by an authorized administrator of Digital Commons @ NJIT. For more information, please contact [digitalcommons@njit.edu](mailto:digitalcommons@njit.edu).

# New Jersey Institute of Technology

## CS288 - Intensive Programming in Linux

### Course Syllabus Spring - 2024

**Instructor:** Mohit Dale

**Office:** GITC 4201 A

**Phone:** (973)-596-2655

**Office hours:** Friday 4pm to 5:45pm || by appointment (Webex)

**E-mail:** [mohit.dale@njit.edu](mailto:mohit.dale@njit.edu)

**Class TA:** TBD

### Class Meetings:

**CS288-006:** Wednesday: 8:30AM to 11:20AM- **Location:** KUPF 104

**CS288-008:** Thursday 8:30AM to 11:20 AM - **Location:** CKB 124

**CS288-010:** Friday 8:30AM to 11:20AM - **Location:** CKB 124

**CS288-012:** Friday 1PM to 3:50PM - **Location:** CKB 124

### Course Description:

This comprehensive program is designed to provide students with a strong foundation in programming concepts and practical skills within the Linux environment. Over the duration of the course, we will cover a wide range of topics such as Bash, C programming, and Linux system API; methods and algorithms for processing web data, such as searching trees and matrix computing; end-to-end applications such as one that constantly presents top 100 stocks; and extending the applications to run on multiple machines to equip you with the necessary tools and knowledge to become proficient in the Linux environment.

### Course Outcome:

Upon completion of the course, students will be able to:

- Have a solid understanding of the Linux command line interface and be able to navigate, manipulate files, and execute commands efficiently.
- bash scripting, enabling you to automate tasks and create powerful scripts.

- Be proficient in using regular expressions, a fundamental skill for pattern matching and text processing in Linux.
- Gain the ability to debug C programs effectively, employing various debugging techniques to identify and fix issues in your code.
- Understand the binary representation of data and be able to perform bitwise operations, enabling you to work with binary data efficiently.
- Learn radix-sort, a powerful sorting algorithm, and apply it to efficiently sort.
- C pointers and dynamic multi-dimensional arrays, essential for managing memory and manipulating complex data structures.
- Explore self-referential structures and linked lists, fundamental concepts in data structures and algorithms, and gain the ability to implement them in C.
- Acquire knowledge and skills in A\* search, a widely used algorithm for pathfinding and problem-solving.
- Develop proficiency in Linux file operations and directory operations, allowing you to create, modify, and manage files and directories programmatically.
- Learn how to write programs with multiple threads, enabling you to leverage parallelism and concurrency in your applications.
- Gain insight into parallel computing with MPI (Message Passing Interface), a powerful framework for developing high-performance parallel applications.
- Explore the exciting field of web scraping, learning how to extract data from websites programmatically and utilize it.

## Prerequisites:

- CS 100 Roadmap to Computing
- CS 280 Programming Language Concepts
- We will be using the C programming language in this Course, you need to know how to write C programs.
- Additionally, this Course is all about Linux Systems, you will be using Linux CLI (Command Line Interface).

## Course Material:

We will use powerpoint slides and a collection of papers/ articles / blogs for our class. All the course materials will be available on canvas, a day before the class (Take a look at the Topics before each class that we will be covering).

### **Recommended Book:**

The C Programming Language, Kernighan and Ritchie, Prentice Hall, 2nd edition, ISBN: 978-0131103627.

## Communication:

This course uses Canvas for announcements and discussion. If you have questions about the class materials or assignments, requests for clarification, or other issues that may interest the class as a whole, post them to the Discussion Forum in Canvas.

**If you have any further questions that you are confident do not belong on Canvas, drop me a message using NJIT email.**

**\*A class discord or Slack account can be set up for Queries related homeworks on the request of the students.**

## Grading Policy:

<b>Homeworks</b>	<b>25%</b>
<b>2 - Midterms</b>	<b>40% - (20% per each Exam )</b>
<b>Final</b>	<b>35%</b>
<b>Total</b>	<b>100%</b>

\*Active class participation is Necessary.

## Homeworks:

Homeworks will have several tasks where code needs to be completed. Each Homework will have its own detailed instructions. In addition, own research on the details of the implementation needs to be conducted. Each Homework will be given a week time to be completed and submitted via Canvas. (More details on Homeworks will be given in the future).

## Exams:

All Exams including mid-term and final will be closed book, failure to attend the exam will result in zero for that particular exam and there will be NO make up exams if you miss.

## Late Policy

Assignments due date will be provided for each Assignment on canvas and students are expected to submit on the day of due date, failing to submit before the due date will result in a penalty of 10% deduction for each day late. No Exceptions will be made unless the student genuinely has a serious problem (like medical, family etc.)

## Grading Scale:

Grade	Significance
85-100	A (Excellent)
75-84	B+ (Very Good)
65-74	B (Good)
60-64	C+ (Acceptable)
56-59	C (Marginal Performance)
51-55	D (Minimal Performance)
50 and below	(F) Fail

## Grade Corrections

Check the grades in course work and report errors promptly. Please try and resolve any issue within one week of the grade notification.

## NJIT Honor Code:

Any evidence of cheating in any form, including plagiarism, Submitting other student Homeworks etc, will be dealt with according to the honor code of NJIT (course failure and suspension or expulsion). Please note: There will be no warnings or chances with regard to cheating. Any discovered case of cheating will be immediately passed to the Dean of Students for further investigation. Cheating is not worth it. You may not only fail this course but also be suspended from NJIT. The full text of the NJIT Honor Code is available for your review at: <https://www5.njit.edu/policies/sites/policies/files/NJIT-University-Policy-on-Academic-Integrity.pdf>

A set of ethical principles governing this course:

- It is okay to share information and knowledge with your colleagues/classmates, but
- **It is not okay** to share the code,
- **It is not okay** to post or give out your code to others (also in the future!),
- **It is not okay** to use code from others for Assignment!

## Course Schedule: -

\* **Note:** The course outline and content are Subject to modification.

The syllabus is divided into **3** stages.

Week	Topics
Stage - 1	
1	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Course Introduction</li><li>● <b>Presentation</b>- Introduction to Linux Environment (CLI)</li></ul>
2	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Introduction to Bash Shell Scripting</li></ul>
3	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Pattern Matching with Regular Expressions</li></ul>
4	<ul style="list-style-type: none"><li>● Midterm Exam - 1<ul style="list-style-type: none"><li>○ CS288-006 - 02/07/24</li><li>○ CS288-008 - 02/08/24</li><li>○ CS288-010 - 02/09/24</li><li>○ CS288-012 - 02/09/24</li></ul></li></ul>
Stage - 2	
5	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Debugging in C (GDB Tutorial) (3.1~3.2),</li><li>● <b>Presentation</b>- Binary representation of data (4.7, 4.3, 4.8)</li><li>● <b>Presentation</b>- Bitwise operators (4.6)</li></ul>
6	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Radix-sort</li><li>● <b>Presentation</b>- C Pointer variables and Memory Addresses (2.1~2.6, 2.9.2~2.9.4)</li></ul>
7	<ul style="list-style-type: none"><li>● <b>Presentation</b>- C Pointer variables (Pointer to Pointer) and (Dynamic) multidimensional arrays (2.1~2.6, 2.9.2~2.9.4)</li></ul>
8	<ul style="list-style-type: none"><li>● Midterm Exam -2<ul style="list-style-type: none"><li>○ CS288-006 - 03/06/24</li><li>○ CS288-008 - 03/07/24</li><li>○ CS288-010 - 03/08/24</li><li>○ CS288-012 - 02/08/24</li></ul></li></ul>
9	<ul style="list-style-type: none"><li>● Spring Recess - No class</li></ul>
Stage - 3	
10	<ul style="list-style-type: none"><li>● <b>Presentation</b>- Structures and Linked list (2.7)</li><li>● <b>Presentation</b>- A* Search and State Space Search</li></ul>
11	<ul style="list-style-type: none"><li>● <b>Presentation</b> - Programs with multiple threads (14.1~14.3)</li></ul>

12	<ul style="list-style-type: none"> <li>● <b>Presentation</b> - Parallel computing with MPI (15.2)</li> </ul>
13	<ul style="list-style-type: none"> <li>● <b>Presentation</b>- Web Scraping</li> </ul>
14	<ul style="list-style-type: none"> <li>● <b>Presentation</b>- Web Scraping (Continuation)</li> </ul>
15	<ul style="list-style-type: none"> <li>● Reading Day</li> </ul>
16	<ul style="list-style-type: none"> <li>● Final Exams - See the registrar's page for final date</li> </ul>