**Midterm Study Guide**

**Chapters 1 through 12: The Basics of the Linux Command Line**

**Chapter 1: What Is the Shell?**

* **Key Concepts:**
  + **Command History**: Learn to use the up/down arrow keys to browse through previous commands and history to list past commands.
  + **Cursor Movement**: Use shortcuts such as Ctrl + a to move to the start of a line, Ctrl + e to go to the end of a line.
  + **Exiting a terminal session**: Know how to safely exit using exit or logout.
* **Commands to Know:**
  + exit, history, !! (repeat last command), Ctrl + a, Ctrl + e, Ctrl + r (reverse search through history)

**Chapter 2: Navigation**

* **Key Concepts:**
  + **Understanding the file system tree**: Recognize the hierarchical structure of directories.
  + **Current Working Directory**: Use pwd to display the present directory.
  + **Changing Directories**: Learn to navigate using absolute and relative pathnames (cd, cd /, cd .., cd ~).
  + **Listing Directory Contents**: Master ls and its options, like ls -l for detailed listings and ls -a to show hidden files.
* **Commands to Know:**
  + pwd, cd, ls -l, ls -a, cd .., cd ~

**Chapter 3: Exploring the System**

* **Key Concepts:**
  + Learn to use ls options for detailed file information (ls -l).
  + **File Type Identification**: Use file to check the type of a file (e.g., text, directory, executable).
  + **Viewing File Contents**: Use cat, less, and head to view contents of files. Know the differences between them.
  + **Symbolic and Hard Links**: Understand the purpose of symbolic links (ln -s) and hard links (ln).
* **Commands to Know:**
  + ls -l, file, cat, less, head, ln -s, ln

**Chapter 4: Manipulating Files and Directories**

* **Key Concepts:**
  + **Wildcards**: Learn to use \*, ?, and [] for file matching.
  + **File Manipulation**: Practice copying (cp), moving/renaming (mv), and removing (rm) files and directories.
  + **Creating Directories**: Use mkdir to create directories.
  + **Linking Files**: Practice creating both **hard links** (ln) and **symbolic links** (ln -s).
* **Commands to Know:**
  + cp, mv, rm, mkdir, ln, ln -s

**Chapter 5: Working with Commands**

* + **Key Concepts:**
    - **Command Identification**: Use which and type to determine if a command is a shell built-in or an executable.
    - **Command Documentation**: Master the use of man pages, help, and --help for command documentation.
    - **Aliases**: Learn to create shortcuts for commands using alias.
  + **Commands to Know:**
    - type, which, man, apropos, whatis, alias, unalias

**Chapter 6: Redirection**

* **Key Concepts:**
  + **Redirection**: Redirect standard output (>), and standard error (2>).
  + **Pipes**: Use pipes (|) to send the output of one command as input to another.
  + **Filters**: Use commands like grep, wc, uniq to filter and manipulate text streams.
* **Commands to Know:**
  + >, >>, 2>, |, cat, grep, wc, uniq, head, tail, tee

**Chapter 7: Seeing the World as the Shell Sees It**

* **Key Concepts:**
  + **Expansion**: Learn how the shell expands commands using wildcards and variables.
  + **Pathname Expansion**: Use wildcards like \*, ?, and [].
  + **Tilde Expansion**: Understand the use of ~ for referencing the home directory.

**Chapter 8: Advanced Keyboard Tricks**

* **Key Concepts:**
  + **Command-Line Editing**: Learn to navigate and edit commands using keyboard shortcuts (Ctrl + a, Ctrl + e, etc.).
  + **History Commands**: Use history to view previously executed commands and !n to repeat a command.
  + **Command Completion**: Use Tab for auto-completion of commands and file names.
* **Commands to Know:**
  + Ctrl + a, Ctrl + e, Ctrl + r, history, !n, Tab

**Chapter 9: Permissions**

* **Key Concepts:**
  + **File Ownership**: Understand the difference between owner, group, and others.
  + **File Permissions**: Recognize the three permission types (read, write, execute) and how to set them using chmod (symbolic and numeric).
  + **Changing Ownership**: Use chown to change file ownership.
* **Commands to Know:**
  + chmod, chown, sudo, su

**Chapter 10: Processes**

* **Key Concepts:**
  + **Viewing Processes**: Use ps and top to view running processes.
  + **Managing Processes**: Learn how to control processes with job control (bg, fg, jobs) and send signals to terminate processes (kill, killall).
  + **Foreground/Background**: Move processes between the foreground and background.
* **Commands to Know:**
  + ps, top, kill, killall, bg, fg, jobs

**Chapter 11: The Environment**

* **Key Concepts:**
  + **Environment Variables**: Learn about important environment variables like PATH, HOME, USER.
  + **Modifying the Environment**: Use export to set environment variables.
  + **Startup Files**: Understand .bashrc and .bash\_profile and how to customize the environment using these files.
* **Commands to Know:**
  + env, export, printenv, source, set

**Chapter 12: A Gentle Introduction to Vi**

* **Key Concepts:**
  + **vi Basics**: Understand the two modes of vi—insert mode and command mode.
  + **Cursor Movement**: Learn to move the cursor within the document.
  + **Basic Editing**: Know how to save (:w), quit (:q), and force quit without saving (:q!).
  + **Text Deletion and Insertion**: Use dd, yy, p, and u for deletion, copying, and undo.
* **Commands to Know:**
  + vi, :wq, :q!, i, dd, yy, p, u, Esc

**Chapters 16 through 20: Intermediate Command Line Usage**

**Chapter 16: Package Management**

* **Key Concepts:**
  + **Installing and Removing Packages**: Know how to install, remove, and update software packages using apt.
* **Commands to Know:**
  + apt-get install, apt-get remove, apt-get update

**Chapter 17: Storage Media**

* **Key Concepts:**
  + **Viewing File Systems**: Use df to view mounted filesystems and du to check disk usage.
  + **Device Names**: Understand how to determine device names using lsblk or fdisk.
* **Commands to Know:**
  + mount, umount, df, du, lsblk, fdisk

**Chapter 18: Networking**

* **Key Concepts:**
  + **Basic Network Diagnostics**: Use tools like ping and traceroute to diagnose network connectivity.
  + **Secure Communication**: Use SSH (ssh) to securely connect to remote machines.
  + **File Transfers**: Use scp to transfer files between computers.
* **Commands to Know:**
  + ping, traceroute, ifconfig, ssh, scp

**Chapter 19: Searching for Files**

* **Key Concepts:**
  + **File Search Tools**: Learn to use locate and find to search for files based on criteria such as name, size, modification time.
* **Commands to Know:**
  + find, locate, fzf, ripgrep

**Chapter 20: Archiving and Backup**

* **Key Concepts:**
  + **Compressing Files**: Know how to compress files using gzip, bzip2, and zip.
  + **Archiving**: Use tar to create and extract tarballs.
* **Commands to Know:**
  + tar, gzip, bzip2, zip

**Chapters 24 through 35: Advanced Shell Scripting and Automation**

**Chapter 24: Writing Your First Script**

* **Key Concepts:**
  + **Script Structure**: Understand the basic structure of shell scripts, including the shebang (#!/bin/bash) and executable permissions.
  + **Script Location**: Know where to store and organize scripts.
* **Commands to Know:**
  + #!/bin/bash, chmod +x, ./scriptname

**Chapter 25: Starting a Project**

* **Key Concepts:**
  + - **Variables**: Understand how to declare and use variables in shell scripts.

**Chapter 26: Top-Down Design**

* **Key Concepts:**
  + **Functions**: Learn how to write and call functions in shell scripts.
  + **Local Variables**: Use local to create function-specific variables.
* **Commands to Know:**
  + function\_name(), local

**Chapter 27: Flow Control: Branching with if**

* **Key Concepts:**
  + **Conditional Statements**: Use if, then, else, and elif for conditional execution of code.
  + **Testing Conditions**: Use the test command and square brackets ([ ]) to evaluate expressions.
* **Commands to Know:**
  + if, then, else, elif, fi, test, [ ]

**Chapter 28: Reading Keyboard Input**

* **Key Concepts:**
  + **Reading User Input**: Use read to prompt the user for input and store values in variables.

**Commands to Know:**

* + - read

**Chapter 29: Flow Control: Looping with while/until**

* **Key Concepts:**
  + **Loops**: Use while loops for repetitive tasks, and until loops for repeating tasks until a condition is met.
  + **Breaking Loops**: Use break and continue to control the flow within loops.
* **Commands to Know:**
  + while, until, do, done, break

**Chapter 31: Flow Control: Branching with case**

* + **Key Concepts:**
    - **Case Statements**: Use case to simplify branching logic when dealing with multiple conditions.
    - **Patterns**: Understand how to use patterns and wildcards in case.
  + **Commands to Know:**
    - case, esac

**Chapter 32: Positional Parameters**

* **Key Concepts:**
  + **Positional Parameters**: Use $1, $2, etc., to access command-line arguments in scripts.
  + **Shifting Parameters**: Use shift to move positional parameters within a script.
* **Commands to Know:**
  + $1, $2, shift

**Chapter 33: Flow Control: Looping with for**

* **Key Concepts:**
  + **For Loops**: Use for loops to iterate over a list of items.
  + **C-style for loops**: Learn to use a more complex for loop with C-style syntax.
* **Commands to Know:**
  + for, in, do, done

**Chapter 34: Strings and Numbers**

* **Key Concepts:**
  + **String Manipulation**: Use parameter expansion to manipulate strings (substring extraction, length, etc.).
  + **Arithmetic Operations**: Perform arithmetic calculations using $(( )).
* **Commands to Know:**
  + ${var}, $((expression)), expr, bc

**Chapter 35: Arrays**

* **Key Concepts:**
  + **Arrays**: Learn how to declare arrays, access array elements, and perform operations on arrays.
  + **Associative Arrays**: Understand associative arrays and their usage.
* **Commands to Know:**
  + array\_name=(value1 value2), ${array\_name[0]}, ${#array\_name[@]}, declare -A