



# UNIVERSITY EXAMINATIONS

**SECOND SEMESTER 2023/2024 ACADEMIC YEAR**

**FOURTH YEAR EXAMINATION FOR THE DEGREE OF  
BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

**COMP 425: UNIX OS AND PROGRAMMING**

***STREAM: R***

***TIME: 2 HRS***

***DAY: THURSDAY [8.30 – 10.30 A.M]***

***DATE: 11/04/2024***

**THIS QUESTION PAPER CONSISTS OF FOUR (4) PAGES**

**PLEASE DO NOT OPEN UNTIL THE INVIGILATOR SAYS SO.**



**INSTRUCTIONS: Section A is compulsory and has 30 marks.**

Attempt any **TWO** questions from Section **B**. Each question has **20 marks**.

**SECTION A – 30 MARKS (ANSWER ALL QUESTIONS)**

**QUESTION ONE (30 MARKS)**

- a. Write the Unix commands to perform the following tasks **(4 Marks)**
  - i. Change the current directory to a directory named "documents."
  - ii. List all files in the current directory, including hidden files.
  - iii. Display the contents of a file named "example.txt" using the **cat** command.
  - iv. Create a new directory named "backup" within the current directory.
- b. Write a simple shell script that does the following: **(4 Marks)**
  - i. Prompts the user to enter their name.
  - ii. Prints a greeting message with the user's name.
  - iii. Checks if a file named "output.txt" exists. If it does, append the greeting message to the file; otherwise, create the file and write the greeting message.
- c. Explain the difference between a foreground process and a background process in Unix. **(2 Marks)**
- d. Demonstrate how to send a running process to the background and bring it back to the foreground. **(3 Marks)**
- e. Use the '**ps**' command to list all processes, and filter the output to only show processes related to a specific user. **(3 Marks)**
- f. Write a Unix command using **grep** to search for lines in a file "log.txt" that contain the word "error." **(4 Marks)**
- g. Define the following terms related to Unix Programming **(3 Marks)**
  - i. Shell
  - ii. Piping
  - iii. Kernel
- h. Provide a brief explanation of the term "signal" in Unix and give an example of a situation where you might use the SIGINT signal. **(4 Marks)**
- i. Differentiate the following terms related to Unix Programming **(3 Marks)**

- i. inter-active(background)process and non-interactive(background) process
- ii. runaway process and Daemon Processes
- iii. Spell and Ispell command

## SECTION B – 40 MARKS (ANSWER ANY TWO QUESTIONS)

### QUESTION TWO (20 MARKS)

- a. **(10 Marks)**
- i. Discuss the Unix commands and tools available for controlling and monitoring processes. Explain how to start a process in the background, stop a running process, and check the resource utilization of a specific process.
  - ii. Provide examples of how you might use commands like ps, top, and kill to manage and monitor processes
- b. **(10 Marks)**
- i. Imagine a scenario where a Unix server is facing performance issues, including high CPU utilization. Discuss the steps you would take to identify the root cause of the problem.
  - ii. Propose solutions to optimize system performance, considering factors such as process prioritization, memory management, and disk I/O.

a.

### QUESTION THREE (20 MARKS)

- b. Outline security measures that should be considered when developing network applications on Unix. Discuss the use of encryption, authentication, and access controls to protect sensitive data during transmission. **(6 Marks)**
- c. Write a shell script that automates the process of regular backups for critical system files. The script should create a compressed archive of specified directories and store them with a timestamp in a designated backup directory. **(10 Marks)**
- d. Describe two advantages of using shell scripting to automate file backups in Unix compared to manual backups. **(2 Marks)**

- e. Write a basic shell script that copies all files with the extension *.txt* from the */data* directory to a backup directory named */backups* on the same system. Assume the backup directory already exists. **(2 Marks)**

#### QUESTION FOUR (20 MARKS)

- a. Define shell scripting and its importance in Unix programming. Provide an example of a simple shell script that takes user input, performs a calculation, and outputs the result. Discuss how shell scripts enhance automation in Unix environments. **(10 Marks)**
- b. Explain the purpose of the `fork()` system call in Unix. Provide a simple code example illustrating the use of `fork()` to create a child process. **(6 Marks)**
- c. Briefly explain why directly using the `cp` command with the `-r` flag might not be the desired approach. **(1 Mark)**
- d. Write a shell script snippet using a loop to achieve the recursive copy functionality. You can use any standard loop construct **(3 Marks)**

#### QUESTION FIVE (20 MARKS)

- a) You are tasked with automating a file management system using shell scripting. **(20 Marks)**
- i. Write a shell script that scans a specified directory for files older than 30 days and moves them to an "archive" directory. Ensure the script logs the moved files and their timestamps.
  - ii. Extend the script to encrypt all files in the "archive" directory using the `gpg` command. Allow users to decrypt files on demand. Include error handling and user prompts for password input during encryption and decryption processes.