National University of Computer and Emerging Sciences

Operating Systems Lab (CL2006)

Sessional-I Exam (B)

Date: March 20th 2024

Course Instructor(s)

Ms. Mubashra Fayyaz
Ms. Fatima Gado
Mr. Sameer Faisal
Mr. Nauman Rajpoot

Total Time: 3 Hours

Total Marks: 25

Total Questions: 03

Semester: SP-2024

Campus: Karachi

Department(s): AI, CS, SE, CY

_________________________ ________ ________ _____________________

Student Name Roll No Section Student Signature

CLO # 1: Understand and Analyze Command Line tools for Linux OS and Shell scripts for system level programming to automate tasks such as file management, system backups and software installations. (Lab # 1 and Lab # 3)

Q1. [9.5 marks]

(a) Write a shell script to customize your own operating system that gives a menu to the user to do the following stuff: **(Use only while loops wherever required)**

1. Create text file(s) with name(s) to be input by user.
2. Open text file(s) to edit. Name(s) of the file(s) to be input by user.
3. Change permissions of a file (name input by user) to all.

Handle all possible errors.

(b) Write a shell script to manage company records. Your script should allow the user to input the following details for the company using command line arguments (separated by space): Company name, and Names and Salaries of 3 employees. Once the data is entered, display a menu with the following options:

1. Calculate the average salary of the company.
2. Display the employee with the highest salary.
3. Sort employees by salary and display the sorted list along with employee name
4. Save employee records to a text file and view the text file.
5. Exit the program.

Upon selecting an option, your script should perform the corresponding operation and display the result. Ensure that your script handles invalid inputs gracefully and provides appropriate error messages.

CLO # 2: Gain hands on experience in writing code that interacts with operating system services related process and files system, multi-thread programing and different synchronization primitives. (Lab # 4 and Lab # 6)

Q2. [9.5 marks]

(a) The Collatz conjecture concerns what happens when we take any positive integer n and apply the following algorithm:

$$n = \begin{cases} n/2, & \text{if n is even} \\ 3 \times n + 1, & \text{if n is odd} \end{cases}$$

The conjecture states that when this algorithm is continually applied, all positive integers will eventually reach 1. For example, if n = 35, the sequence is 35, 106, 53, 160, 80, 40, 20, 10, 5, 16, 8, 4, 2, 1. Write a C program using the fork() system call that determines the length of the Collatz sequence until it reaches 1. The starting number will be provided from the command line. For example, if 8 is passed as a parameter on the command line, the child process will output 8, 4, 2, 1. Because the parent and child processes have their own copies of the data. Have the parent invoke the wait() call to wait for the child process to complete before exiting the program. Perform necessary error checking to ensure that a positive integer is passed on the command line.

(b) An echo server echoes back whatever it receives from a client through shared memory, after reversing each character in the message. For example, if a client sends the server the string “operating system”, the server will print “metsys gnitarepo”.

CLO # 3: Understand how to configure and customize Linux Kernel for installations, applying patches and performance optimizations. (Lab # 2)

Q5. [6 marks]

(a) Create a geometric calculator program using three files: **main.c**, **geometry.c**, and **geometry.h**. In **main.c**, utilize functions from **geometry.c** to compute the area and perimeter of basic geometric shapes like circles, rectangles, and riangles. Develop a straightforward Makefile, for compiling C source files into object files and a clean rule to remove object files and the executable. In **geometry.c**, implement functions for area and perimeter calculations for circles, rectangles, and triangles. Submit the completed C files, header files, Makefile, and snapshots of the terminal output.

(b) Linux Recompilation
The general steps to recompile the Ubuntu kernel are:

1. Installing dependencies
2. Download Kernel Sources
3. Configure Kernel (Write command for this):

   ____
4. Compile Kernel
5. Install Modules (Write command for this):

   ____
6. Install New Kernel
7. Update GRUB Configuration
8. Reboot (Write command for this):

   ____
9. Verify