CSE2005-Operating Systems Lab

Assessment-1 Questions

(a) Shell Programming

- Handling the command line arguments
- String reversal
- If-Else, Nested If Else,Switch cases in shell

(b) Parent child process creation using fork( ) and exec() system call

Checking the Process Identifier
Assigning new task to child
Providing the path name and program name to exec()
Synchronizing Parent and child process using wait()

(c) Process and Thread Management

Write a program to create a thread and perform the following **(Easy)**

- Create a thread runner function
- Set the thread attributes
- Join the parent and thread
- Wait for the thread to complete

(d) Write a program to create a thread to find the factorial of a natural number 'n'. **(Medium)**

(e) Assume that two processes named client and server running in the system. It is required that these two processes should communicate with each other using shared memory concept. The server writes alphabets from a..z to the shared memory .the client should read the alphabets from the shared memory and convert it to A…Z. Write a program to demonstrate the above mentioned scenario. (**Medium)**

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

n = n/2, if n is even n = 3 × n + 1, if n is odd

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 generates this sequence in the child process. 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, it will be necessary for the child to output the sequence. Have the parent invoke the wait () call to wait for the child process to complete before exiting the program **(High)**.

g) Write a multithreaded program that calculates various statistical values for a list of numbers. This program will be passed a series of numbers on the command line and will then create three separate worker threads. One thread will determine the average of the numbers, the second will determine the maximum value, and the third will determine the minimum value. For example, suppose your program is passed the integers 90 81 78 95 79 72 85 , the program will report the average value as 82. The minimum value as 72. The maximum value as 95. The variables representing the average, minimum, and maximum values will be stored globally. The worker threads will set these values, and the parent thread will output the values once the workers have exited. **(High)**