**Operating Systems Lab 2022**

**Assignment Set 4 :**

**Submission Deadline 19th Oct 2022**

**------------------------------------------------------------------------------------**

**Objectives: to learn the following:**

* **Use of pthread library**
* **Creation of attachable & Detachable thread**
* **Learn the use of ps-eLF, top, ipcs –m, ipcs –m commands**
* **Shared memory creation and use**

A1) Observe and compare the physical time taken to (i) create 100,000 child processes (ii) create 20,000 child threads. Write two separate codes to perform the experiment. Use some system call to read the system clock time just before the creation starts and just after all the creation is over, display the difference in physical time.

Check if there is any significant speedup achieved in using threads which are light-weight processes.

================================================================

A2) Write a suitable code using **clone() system call** to speedup finding all prime numbers in a given range [1, N], creating M child threads (LWPs) by the parent process thread.

First try to run it for two threads then generalize it for M child threads

================================================================

A3) Solve the **Assignement 2 by using pthread library functions** instead of clone() system call.

================================================================

A4) Write a program using **pthread library** functions in which a parent process thread creates n child threads, such that even numbered child threads are created in attachable mode and odd numbered child threads are created in detachable mode. Show that the child threads created in attachable mode communicates by passing any value (integer) back to the parent using pthread\_exit() function call.

A5) Write a program in which a parent process creates a **shared memory** and puts an integer variable with initial value 1. Next, the parent creates ‘n’ number of children and attach each children with the shared memory. The objective of each child will be to read the integer variable in the shared memory and increment it by one ‘m’ number of times separately. The parent process will synchronize to the termination of all child processes and finally reads the final value of the shared integer variable and displays the value.

Run the program 10 times and observe if the value is equal to consistent each time it got executed?

A6) 1. Write a program in which a parent process will create a **shared memory** in which an array of 10 integers will be there (the initial value of the array elements will be 0). The parent will create a child and attach with the shared memory. The parent will fill up the odd indexed locations of the array by generating the random number (in between 1 to 100). The child will check the odd indexed locations and if it finds it as a non-zero integer then it increments the integer by 2 and store in the next adjacent even indexed location. Finally, the parent process will print the summation and average of integers stored in the even indexed locations (i.e., the integer that are included by the child process) and the child process will do the same for the integers stored in the odd indexed locations (i.e., the integer that are included by the parent process).