

**Department of Computing**

**CS 330: Operating Systems**

**BSCS: 7C**

**Lab 9**: **Programming Threads**

**CLO4(Develop programs to interact with OS components through its API)**

**Date: 15-11-2019**

**Time: 02:00 PM – 05:00 PM**

**Instructor: Dr. Fahad Javed**



CS330: Operating Systems Page 1



**Lab 9: Programming Threads**

**Introduction**

The purpose of this lab is to learn threads programmatically in more depth and learn to ensure safe global memory access.

**Objectives**

Objective of this lab is to enable students use threads in programs and synchronization primitives.

**Tools/Software Requirement**

Linux OS installed on laptops or systems.

**Description**

In order to understand the concept of threads and role of synchronization primitives to ensure safe global variable access you are supposed to Write a program using pthreads, which calculates the sum of elements in a hard-coded integer array in parallel using 4 threads. The program must divide the work between 4 threads which run simultaneously. For simplicity, you can assume that the size of the array is 100. Note that the integer array must be declared as a global data structure.

**Tasks**

1. Initially code your solution so that the sum of elements is maintained in a global shared variable. Each thread modifies the same shared variable as it sums up elements from the array. Use a suitable synchronization primitive to ensure safe access to the global variable.
2. Now modify your solution so that it does not require any synchronization primitives. An idea here is that each thread must first find its own local sum first, and later the main () function can use these local sums to determine the global sum for the entire array.
3. Which approach do you think would run faster, and why?

**Deliverables**

Submit the code for both of your programs, along with the answer to the last question.



CS330: Operating Systems Page 2