Lab1-Multithreading 1

Study and review the code developed during the lecture and posted on Google classroom under Codes folder, then the programs to do the following: For all the programs, you must use chrono library to measure the execution time of the program and display it at the end of the program.

1- Write a program to create a thread to execute a function named myFunc that takes one argument N of type int. The function prints the threadID, the value of the input argument, then sleeps N sec. Your program prints the threadID of the main function as well.

2- Write another program that has a function that takes two arguments (a pointer to a int variable and an integer N). The function calculates the sum of the odd numbers between 1 and N and save this into the int pointer. Use multithreading to create a thread to calculate different sums in a safe way without using mutex or semaphore (we did not study them yet).

3- Write a program to test the effect of iostream random output behavior. Solve the problem of this by using the lock and unlock with cout stream.

4- Assume that we have an array of size N. We would like to multiply each element in the array by (index +1). It is required from you to create a multithreaded program that creates M threads and distributes the tasks to each thread and calculates correct results. Hint: use a temporary array of size M so that each thread save its sum in a cell of that array. Then the main thread must sum all the results to find the final sum. You must handle join carefully. You do NOT have to use mutex or semaphore. Test the performance of the program and compare the running time with the sequential version. Note that it you have to choose N and M so than N=K x M, where K is a constant >1.

Good luck

Dr. Samir Elsagheer