National University of Computer and Emerging Sciences, Lahore Campus

Course:	Operating Systems	Course Code:	CL220
Program:	BS(Computer Science)	Semester:	Spring 2021
Deadline:	29-May-2021	Total Marks:	10
Type	Homework 2	Weight	-
		Page(s):	1

Important Instructions:

1. For submission, name your solution file as your roll number, i.e., 19_1111.cpp, and submit it in homework 2's submission folder on Google classroom. Do not zip your file.
2. You are not allowed to copy solutions from other students. We will check your code for plagiarism using plagiarism checkers. If any sort of cheating is found, negative marks will be given to all students involved.
3. Late submission of your solution is not allowed

Question 1: Calculating the average of a list of integers by exploiting parallelism:

Suppose we have an array of integers: 1, 10, 9, 100, 23, 4, 11, 12, 3, and 1. The size of this array is 10. Now if we want to calculate the average, we can simply do:
average = sum of elements/ total elements.

Now, if we have a large number of elements, then it will take a lot of time to calculate the average. We can utilize parallelism by dividing the task of finding the sum among different threads. Suppose a thread t1 finds the sum of first 4 elements:
sum_1=1+10+9+100= 120
count=4

And another thread t2 calculates the sums of remaining 6 elements:
sum_2=23+4+11+12+3=53
count=6

Then average calculated as:
Total Sum = sum_1+sum_2= 120+53= 173
Total Count= $count_1$ + $count_2$= 4+6=10
Average= Total Sum/Total Count= 173/10= 17.3

Assume that we want to compute the average of a list of numbers. The numbers are not in a single file, but they are placed in several files. Now write a program that is passed the names of files containing integers as command line arguments. The program will then create as many threads as the number of files. Each thread will be passed as the name of a file as an argument. All threads will then work in parallel to compute the sum of all the numbers in the file passed to them as argument. Each thread will then return the sum and the count of numbers to the main thread. The main thread will then compute the average as given below and print the average on the screen.

TotalSum= sum_1+ sum_2+ sum_3+……+sum_n

TotalCount=$count_1$+$count_2$+$count_3$+…….+$count_n$

Average= TotalSum/TotalCount