COLLEGE OF ARTS AND SCIENCES
UNIVERSITY OF THE PHILIPPINES LOS BAÑOS
1/F Wing C Physical Sciences Bldg., Harold Cuzner Royal Palm Ave.
College 4031, Laguna, Philippines
Phone +63 49 536 2313
ics.uplb@up.edu.ph www.ics.uplb.edu.ph
INSTITUTE OF COMPUTER SCIENCE

CMSC 180: Introduction to Parallel Computing
Second Semester 2022-2023

Laboratory Exercise 02 PART 01
Runtime-efficient Threaded Interpolating elevation

Research Activity: Extend the more efficient computer program that you wrote in Laboratory Exercise 01 to use threads to estimate the point elevation of $n \times n$ matrix. In other words, transform your faster serial computer program into a threaded computer program.

Exercise Specifications

A. Write the main program lab02 that includes the following:
   1. Read n and t as user input (maybe from a command line or as a data stream), where n is the size of the square matrix, t is the number of threads to create, and $n > t$
   2. Create a zero $n \times n$ square matrix **M.** Assigned a randomized non-zero value to grid points divisible by 10 such (0,0), (0,10), (10,0), (20,0),(10,10) …… You can use a function for this but the running time of this will not be considered in the *time_elapsed*
   3. Divide your M into t submatrices, $m_1, m_2, ..., m_t$; *You can add an additional filter if the matrix size* ***n*** *is not divisible* ***t*** *such as if n=10 while t =3, then the input values cannot be processed because there is excess column or row.*
   4. Take note of the system $time\_before$;
   5. Create t threads, to interpolate the values for each submatrix. **(IMPORTANT)**
   6. Take note of the system time $time\_after$;
   7. Obtain the elapsed $time\_elapsed = time\_after - time\_before$;
   8. Output the $time\_elapsed$
   9. (Optional) You can output the resulting matrix.

Submit your code through the Google Classroom Laboratory Exercise 02 Part 1 portal.