# **CSC 4760/5750, Spring 2022**

# **Programming Assignment 2: Target Practice**

# **Assign Date: January 31, 2022 -- Due: February 9, 2022, 11:59 pm**

# Program: Target Practice

For this assignment you are going to write a parallel version of the provided serial program.

* The program generates a given number of samples consisting of two random numbers between 0 and 1.
* If the sum of squares is less than or equal than 1, the sample counts as a hit. Otherwise, it is a miss.
* After all the samples are classified as a hit or a miss, the result is calculated using the following equation: Result = 4 \* (hits / misses).

## Input:

* Reads the number of threads and total a number of samples from command line arguments.
* The program should be runnable with “./<executable\_name> <num\_threads> <num\_samples>”
* You can assume valid numerical input.

## Output:

* The result
* Use omp\_get\_wtime() to calculate:
  + The total execution time of the program.
  + The execution time of each thread.

## Program 1:

Parallelize the program using OpenMP with each thread running samples equal to its ID x 1000.

You ignore the provided number of samples from the terminal.

## Program 2:

Parallelize the program using OpenMP by manually assigning the samples as evenly as possible across the supplied number of threads. Your program should work properly with any combination of threads and samples. Account for if the num of samples is not a multiple of the number of threads.

## Program 3:

Parallelize the program by using “pragma omp for” Do **not** use the openMP reduction clause.

### Reports and Submission:

Run your code on the cluster with an interactive node with at least 5 cores. Write a short report with your observations on how each program ran, including screenshots from your runs. Did anything strange happen, or any patterns emerge?

Zip your source codes and report and upload to the ilearn.