**CENG342 Project-2**

Due: April 22, 2022

*Penalty for late submission is 10% per day.*

You cannot upload any of assignments of CENG342 to any web page, forum site or send somebody who does homework for money. If you take a similar attitude, you will be in opposition to the law on the protection of personal data and violation of copyrighted verbatim. In addition, disciplinary action will be arranged by the Rules and Regulations Governing Student Disciplinary Actions in Institutions of Higher Education.

This is a teamwork assignment; you are going to work in group of 3 students.

In this project, you are going to implement a parallel image processing algorithm by using threads. You are going to utilize multi-thread parallelism via OpenMP. Image processing tools provide a large range of different algorithms for all kind of circumstances. A simple and effective algorithm is the **Histogram equalization**. It re-adjusts the contrast of an image using the image's histogram.

I have written its sequential code. You need to implement a parallel version of the same algorithm which should give the same output.

The program that takes two inputs; the first for the input image and the second for the output image. Your program should perform a parallel **Histogram equalization** algorithm on the input image and then store the results into a new image.

You need to measure the time for only Histogram equalization and make tables that compares timings of your algorithms with varying number of threads and different schedules for ‘for’ loops. You need to make tables for speed-up and efficiency values of your parallel algorithm for each schedule too.

**What is required?**

1. Report
   * A report that includes your names and surnames and appropriate title and small description.
   * Very short pseudocode of your **parallel** algorithm with at most **12 lines**.
   * Brief explanation: Steps of your parallel algorithm in terms of 4 steps of Foster’s methodology.
     + Furthermore, you should discuss which parallelism you adopted, task or data parallelism, with the reasoning behind it.
   * Tables; for elapsed timings, speed-ups and efficiencies of your parallel algorithm by using 1,2, and 4 threads, if your PC has more number of cores than 4 you can increase the number of threads. For accurate timing please **take average of at least 3 tests.**
2. Code

* Source code named main.c written in C/C++ programming language.

1. You must test your algorithm with the following `for` schedules:

* Default
* (Static,x): test chunk sizes for 1 and 100
* (Dynamic,x) test chunk sizes for 1 and 100
* (Guided,x) test chunk sizes for 100 and 1000

1. Draw speedup charts for every schedule. In your report, there must be 7 speed-up charts; one for each of the schedule. In the conclusion part, you must specify which schedule gives the best time according to your experiments.

**Notes:**

* Working together with other groups is prohibited.
* Submit your codes together with your report (**PDF** **format**) in a zip file named project2-GroupX.zip into AYBUZEM. (Replace X with your group number.)
* One member in each group should upload the exam on behalf of their group. That is, there should be only one submission for each group in the Aybuzem.