

HPC/ Parallel Processing
Project [10 + 2 Marks]

Problem: Mean, Variance, and Standard Deviation

Suppose that the entire population of interest is eight students in a particular class. For a finite set of numbers, the population standard deviation is found by taking the square root of the average of the squared deviations of the values subtracted from their average value. The marks of a class of eight students (that is, a statistical population) are the following eight values: 2, 4, 4, 4, 5, 5, 7, 9

These eight data points have the mean (average) of 5:

$$\mu = \frac{2 + 4 + 4 + 4 + 5 + 5 + 7 + 9}{8} = \frac{40}{8} = 5.$$

First, calculate the deviations of each data point from the mean, and square the result of each:

$$\begin{array}{ll} (2 - 5)^2 = (-3)^2 = 9 & (5 - 5)^2 = 0^2 = 0 \\ (4 - 5)^2 = (-1)^2 = 1 & (5 - 5)^2 = 0^2 = 0 \\ (4 - 5)^2 = (-1)^2 = 1 & (7 - 5)^2 = 2^2 = 4 \\ (4 - 5)^2 = (-1)^2 = 1 & (9 - 5)^2 = 4^2 = 16. \end{array}$$

The variance is the mean of these values:

$$\sigma^2 = \frac{9 + 1 + 1 + 1 + 0 + 0 + 4 + 16}{8} = \frac{32}{8} = 4.$$

and the population standard deviation is equal to the square root of the variance:

$$\sigma = \sqrt{4} = 2.$$

Write a parallel program to calculate the above mean, variance, and standard deviation using the below algorithm:

1. Each process calculates its local sum on its portion of data. If the division result has a remainder, then send this remainder to the last process (the

- process that has the rank "n-1" or let the master process work on it).
2. The master process calculates the mean (dividing the total sum by the size of the elements) and sends it to all processes.
 3. Each process calculates the squared difference on its portion of data.
 4. The master process then calculates the variance (dividing the total squared difference by the size of the elements).
 5. The master process calculates the standard deviation by getting the square root of the variance and prints the results..

Input: array size, array elements

Output: mean, variance, standard deviation

Sample input/output:

Sample1:

Array size: 8

Array elements: 2, 4, 4, 4, 5, 5, 7, 9

Output: Mean = 5.0, Variance = 4.0, Standard deviation = 2.0

Sample2:

Array size: 7

Array elements: 4, 5, 6, 6, 7, 9, 12

Output: Mean = 7.0, Variance = 6.2857, Standard deviation = 2.5071

Requirements

- Implement the problem once using **MPI** and another time using **OpenMP**.
- Compare the performance of both implementations by calculating **parallel time**, **speed up** and **efficiency** for different sizes and different number of cores (double the problem size with the same number of cores, then double the number of cores to build a table with min 10 rows and 10 columns).
- Write the performance evaluation to a separate .pdf file.

Deliverables

- Source code for both implementation each one in separate file (**.c files only**)
- Performance evaluation report (**.pdf file**)

Submission Rules (Read Carefully)

1. Deadline of the Assignment is **22 May 2023**.
2. You can work with team members of any lab.
3. Project is performed in a team of 3 members.
4. Your code should run on any number of processes and make sure to deliver a running code.

5. Deliver only your ".c" files and ".pdf" in a zipped folder and don't include any ".exe" files.
6. Follow the naming convention: ID1_ID2_ID3_Group to name your folder.
7. You are required to deliver your own effort, cheating any part of your code from your colleagues or from the internet will lead to serious consequences.

References:

https://en.wikipedia.org/wiki/Standard_deviation