# Assignment #2

## Problem Specification:

You need to provide the OpenMP C implementation of the *many-body problem* associated with *dimensionless quantities* as stated below. You need to simulate the many-body system and generate the dynamics trajectory. The detailed procedure is mentioned in the following steps:

1. Consider 1000 *spherical* bodies. The mass of each body is *unity*.
2. Place the bodies inside a rectangular cuboid shaped container whose dimensions (length, width, and depth) are 100, 200 and 400 respectively. If during motion, physical contact happens between the body and/or the container wall, then the body will get reflected from the rigid boundary and will be positioned accordingly. Assume the collision as an elastic collision.
3. The initial 3D coordinates of the bodies along with the specifications are provided in the attached file. Assume initially they all are at rest.
4. You can calculate the force acting on a pair of bodies using the force equation $F = \frac{M_1 \times M_2}{r^2}$, where $M_1$, $M_2$ are the masses of the corresponding bodies and $r$ is the Euclidean distance between them.
5. The actions that you need to perform (algorithm) during each time step of the simulation are:

```
a. For each time step $n$
    i. For each particle $i \in \{1, 2, \ldots, 1000\}$
        1. Calculate force acting on $i$ as
           $$\overrightarrow{F^i} = \sum_{i \neq j} \overrightarrow{F_{ij}}$$
        2. Calculate half-step velocity as
```

$$\vec{v}_{\frac{n+1}{2}} = \overrightarrow{v_n} + \frac{\overrightarrow{f_n}\Delta t}{2m}$$

3. Update particle position as

$$\overrightarrow{r_{n+1}} = \overrightarrow{r_n} + \overrightarrow{v_{\frac{n+1}{2}}}\Delta t$$

4. Calculate full-step velocity as

$$\overrightarrow{v_{n+1}} = \overrightarrow{v_{\frac{n+1}{2}}} + \frac{\overrightarrow{f_{n+1}}\Delta t}{2m}$$

ii. Print particle positions in the trajectory file in the provided input format

6. The OpenMP C version of the many-body simulation (name it *many-body-sim-program.c*) should be run on the given input file for a total number of 720,000 simulation steps. Record the positions of the bodies at each 100 time step in an output file namely, *trajectory.txt* (Use binary file format for reduced file size) in a line (separator between two record is newline character).
7. Repeat the step 6 by varying number of threads 1, 2, 4, and 8 (USE SAME SYSTEM FOR THIS VARIATION). Report the scalability of your implementation for varying thread numbers in file *scalability.txt*. For each variation, maintain the log of your simulation in another file (*simulation_log.txt*) with the following details:
   a. System specification (CPU, #Processor, Clock Speed, #Threads used)
   b. Time required for each step
   c. Total simulation time
   d. Any other information
8. Design some graphics visualizer (*graphics-program*) which will read your *trajectory.txt* file as input and will display the visual output. DO NOT UPLOAD

*trajectory.txt*. Keep your *trajectory.txt* file for each thread numbers (1, 2, 4, and 8) ready and with you so that it can be available on demand.

9. Create a folder with your roll no (who will upload) as the folder name. Place your OpenMP code (*many-body-sim-program.c*), graphics visualizer program (*graphics-program*), scalability report (*scalability.txt*), and simulation log (*simulation_log.txt*) in your created folder. You may consider providing an additional readme file. Zip it and upload it in Moodle.
10. One submission from each group is enough no need to submit by everybody.
11. One demo will be arranged after the submission deadline where randomly picked one student from the group will be asked to present/explain. The group marks will be copied to all the members.

Marking scheme (**Total marks 13**):

a. OpenMP C implementation: **7 marks** (*part marking will be decided later*)
b. Graphics design with demo using trajectory file: **2 marks**
**c.** Simulation Log: **1 marks**
d. Presentation: **3**

**All the plagiarism cases will be treated strictly.**