**Report**

**Name:** Arshak Poghosyan

**Course:** CS 140 Mechanics  
 **Section:** B

### **Summary**

This project centered around simulating two-dimensional packing and space management through principles of classical mechanics. I developed a particle-based model in which forces and motion were handled using Newtonian physics. Over the course of several iterations, I enhanced the simulation to address different constraints—ranging from variable particle mass to limited space—and ultimately improved its computational efficiency using a spatial grid approach.

### **Project Progress**

#### **Getting Started**

To begin, I configured the Open Source Physics (OSP) framework within Eclipse and explored several demonstration simulations. These initial exercises gave me insight into how physical systems can be simulated using particles and how the Verlet integration technique updates their positions over time.

#### **Basic Circular Packing (Equal Mass)**

My first implementation involved simulating a group of circular particles that experience mutual attraction and repulsion. All particles were treated as having identical mass. I used a drag factor to simulate energy loss and encourage system stabilization. The simulation visually displayed particle interactions and recorded metrics such as the bounding box dimensions and total potential energy.

#### **Parameter Tuning**

I experimented with a variety of constants for attraction and repulsion, as well as different values for drag and the simulation timestep. By observing changes in convergence speed and particle layout, I was able to assess the sensitivity of the system to these parameters. I saved results for each variation to compare system behavior and efficiency.

#### **Packing with Area-Based Mass**

In the next phase, I altered the model to assign mass based on particle area (mass ∝ πr²). This adjustment created a noticeable difference—larger particles, being more massive, responded more sluggishly to force. This produced more realistic behavior and often led to more stable configurations. I reused previous setups to ensure consistency in comparisons.

#### **Constrained Space Packing (Floor-Planning)**

For this stage, I enforced strict boundary conditions, confining all particles to a fixed-size rectangular region. I revised the Verlet update mechanism to handle collisions with the rectangle's edges. This constraint made it more challenging to achieve a dense, non-overlapping packing, particularly as particle sizes increased.

#### **Performance Improvement via Spatial Partitioning**

The simulation’s complexity initially scaled poorly due to the O(N²) nature of pairwise interactions. To mitigate this, I implemented a spatial grid that divided the simulation space into smaller cells. Close-range interactions were handled directly, while long-range effects were approximated using aggregate "super-particles" based on a cell's center of mass. This optimization greatly improved simulation speed while maintaining visual and physical fidelity.

#### **Final Optimized Floor-Planning**

The final version integrated both the boundary constraints and the grid-based performance enhancement. This version ran efficiently even with a larger number of particles and produced compact, well-organized layouts within the predefined space. It struck a good balance between accuracy and computational efficiency.