**Chris Schiff – HW2 Documentation**

**Important Note**: This simulation is unfinished.

**Instructions for Use**: Press Spacebar to toggle debug draws. These include displaying the calculated point of the most recent collision as a black dot, as well as a line segment drawn at the middle of the screen (which is the origin in the simulation) of the minimum translation vector of the most recent collision. Press Enter to restart the simulation. All of the game objects will be given random positions, velocities, and angular velocities.

-

**Algorithms Used**

**Broad-Phase Collision Detection**: No broad-phase collision detection takes place in the simulation.

**Narrow-Phase Collision Detection**: 2D SAT checks are used for collision detection for squares, triangles, and hexagons. No circles are used in the simulation. An object of a given shape will only check collisions with other objects of the same shape. Once collision is detected, the point of collision is calculated using an algorithm based off of the ATLAS demo found at: <https://github.com/IGME-RIT/physics-determiningCollisionPoint-ConvexHull-2D>. After the point of collision is calculated, collision response is calculated using equations 8 and 9 from Chris Hecker’s article: <http://chrishecker.com/images/e/e7/Gdmphys3.pdf>.

**Tunneling Prevention**: Only very minor tunneling prevention measures are taken, and they are only applied to the screen’s bounds rather than other game objects. To prevent a game object from getting stuck in one of the 4 bounds, it is moved by 0.01 units in the opposite direction of the bound.

**Linear and Angular Momentum**: Equations 8a and 8b, respectively, from the aforementioned article by Chris Hecker.