

# REFERENCES



VectorStock®

VectorStock.com/23571708

VectorStock®

VectorStock.com/2524

VectorStock®

VectorStock.com/24844542

# SYSTEMS

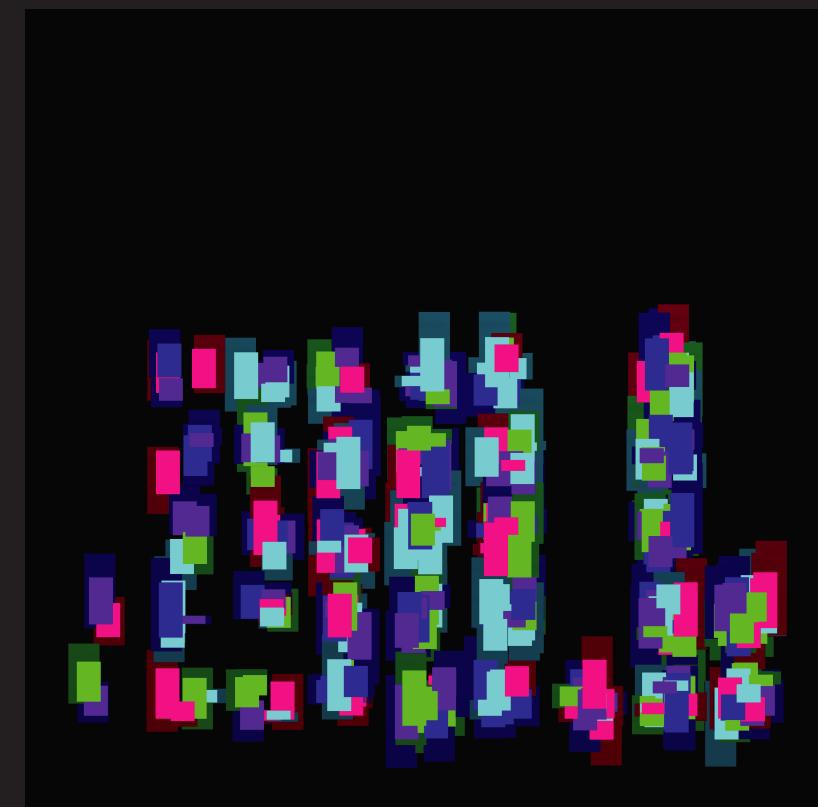
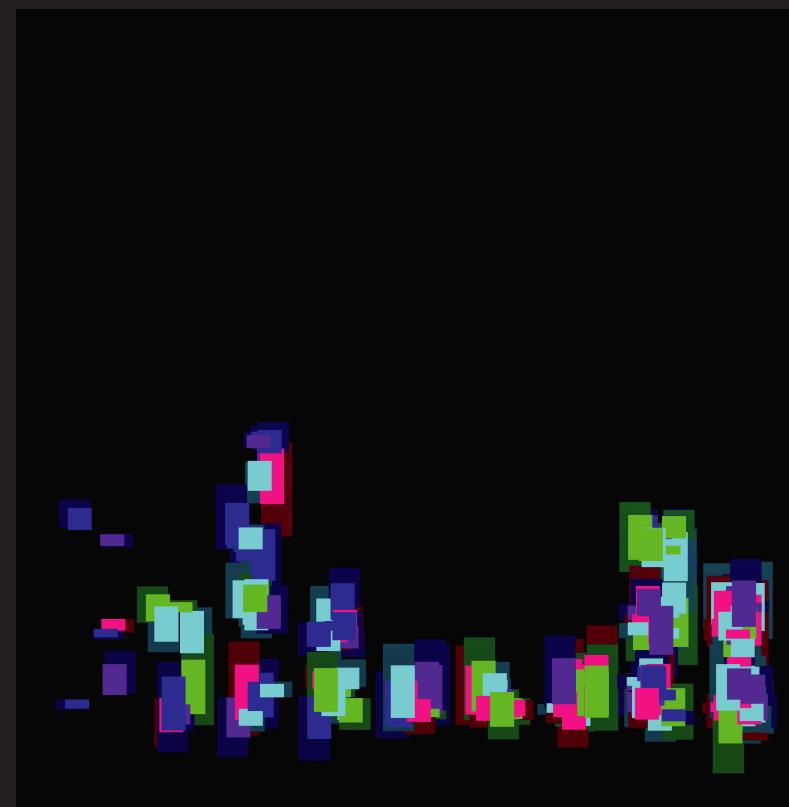
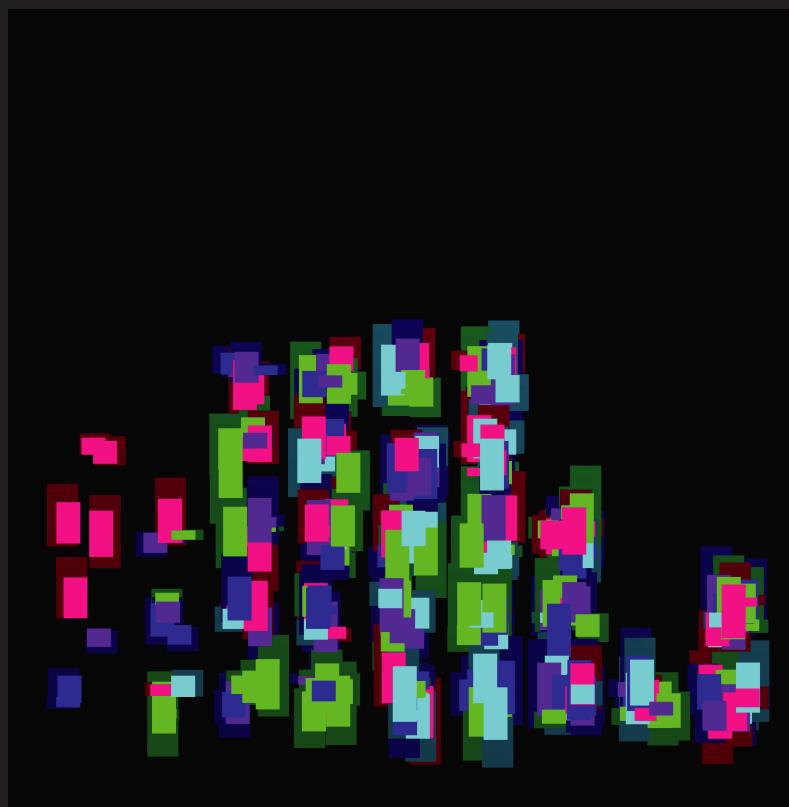
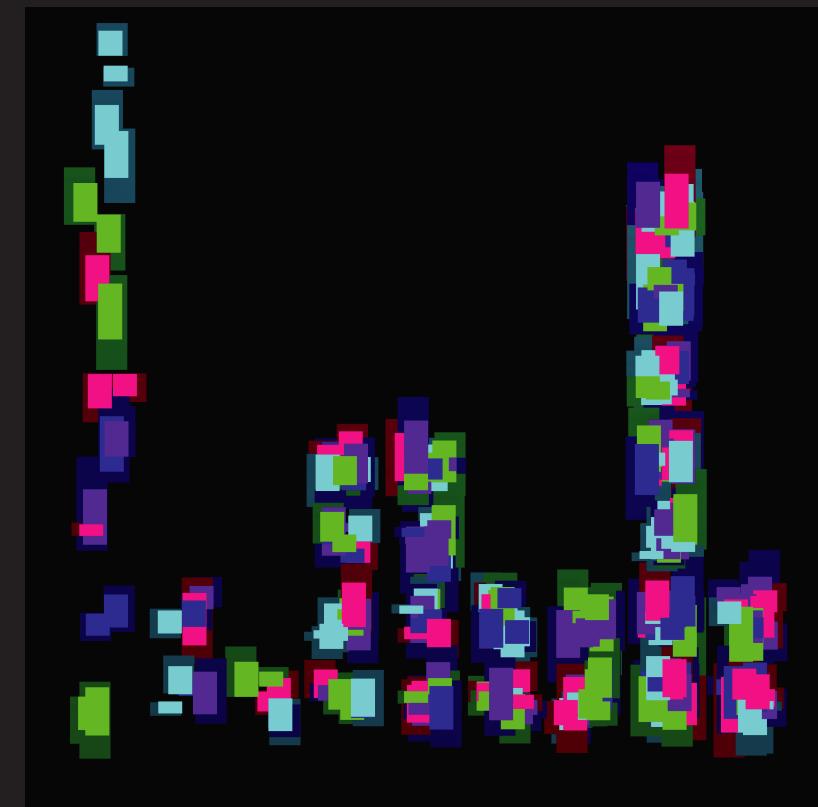
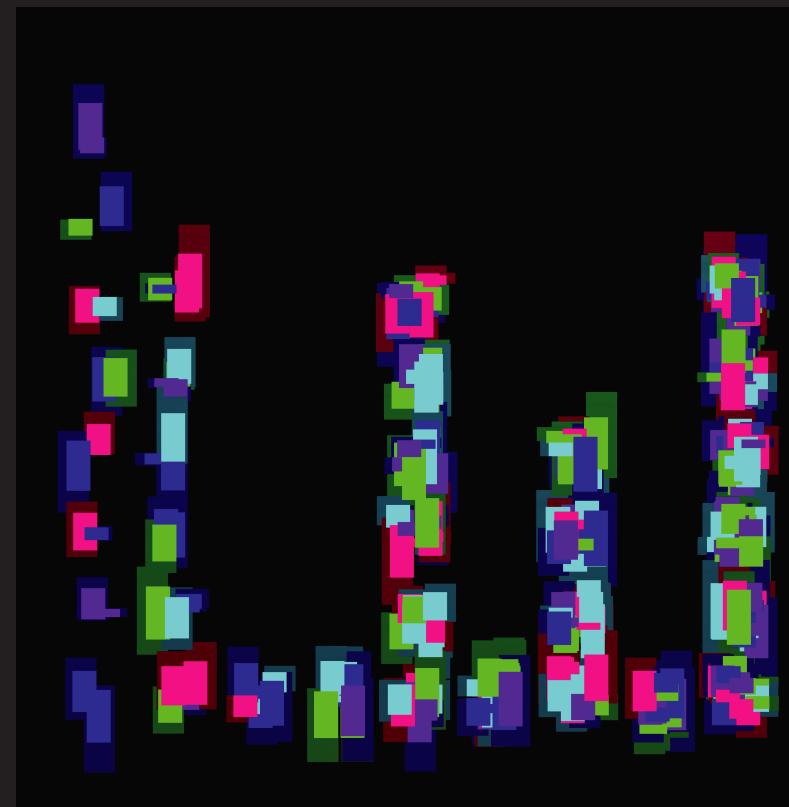
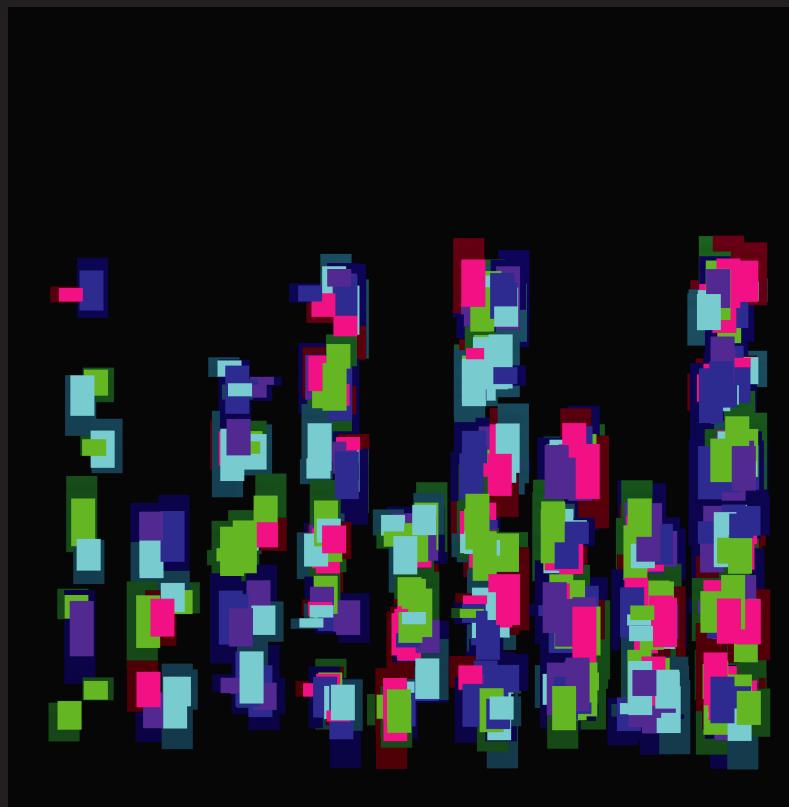
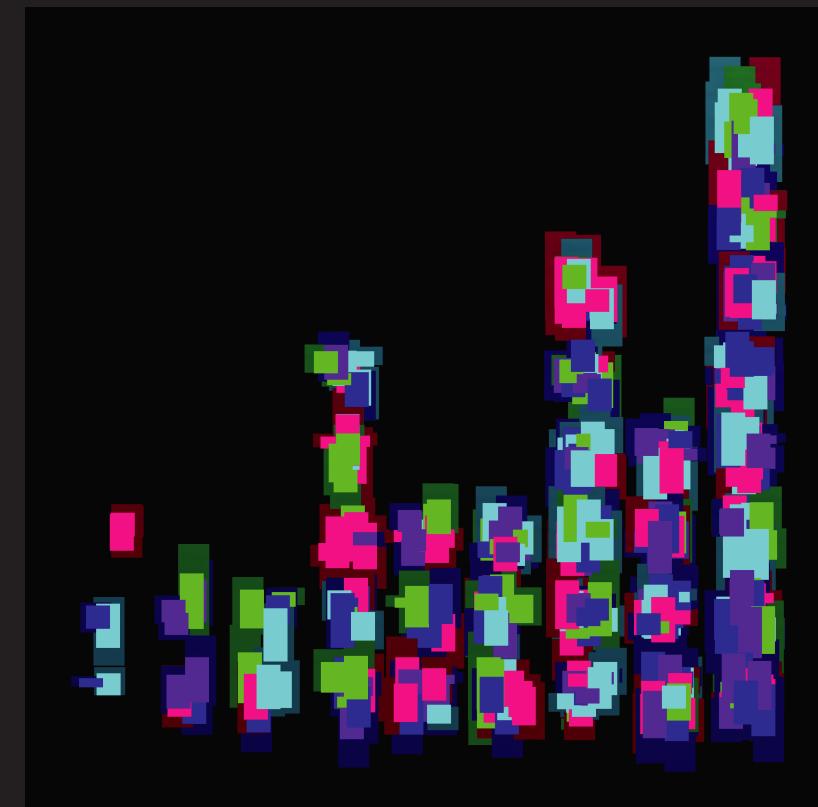
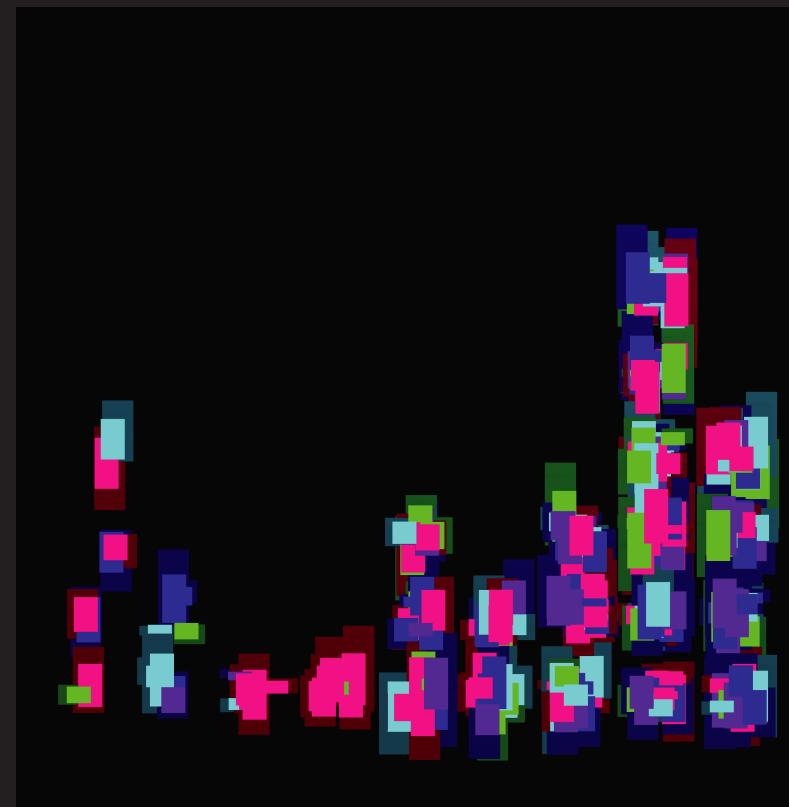
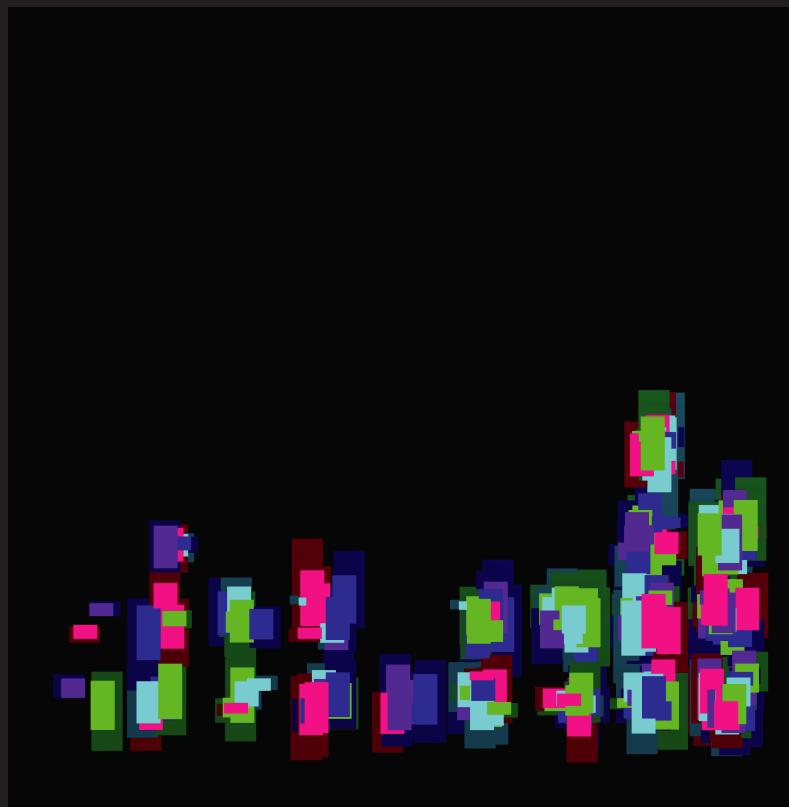
Across all three pages, I utilize the same object with the same behaviors. My object is the single parent to two game objects ("juice" and "juice shader"). The amount of the game objects under the parent are the same, the only difference is that the "juice shader" game object has a darkened color palette and is positioned behind the "juice" game object to give some visual depth.

For each system, what changed was the usage of the manager of my object.

On the first page, the amount of columns in each grid stays the same, but the amount of rows is randomized. Progressing from columns left to right, each object ("glob") becomes increasingly complex, with more game objects generated in each glob.

On the second page, the manager instantiates a single glob in order for subsequent globs to be placed around in a circular fashion. Additional globs were generated along each fraction of the circular grid, with noise and a Sin function implemented to give some curvature to the "arms".

On the third page, the objects within each glob were placed randomly inside of a unit sphere. Then, similar to the second page, the manager instantiates a single glob for subsequent globs to be placed around. There are three separate for loops for three separate instantiations: the first instantiation of globs (in a circle) was of a radius randomly decided in the range, radiusMin to radiusMax (inputted by the user). The second circle was of that radius \* 2, and the third circle was of that radius \* 3, hence the three rings of globs.



# PAGE 2

