**Spike:** Task 26

**Title:** Collisions

**Author:** Sam Huffer, 101633177

# Goals / deliverables:

* At least box-based (axis-aligned rectangles) intersection testing, and circle-circle intersection testing
  + Display at least two boxes, one in a fixed position, the other(s) moving. Detect collisions using axis-aligned rectangle testing and visually display the “collided” status to the user
  + Display at least two circles, one in a fixed position, the other(s) moving. Detect collisions using circle-circle collision testing, and visually display the “collided status to the user.
* Incorporate extensibility in your collision data structure, so that you would be able to swap out one collision detection method for another.

# Technologies, Tools, and Resources used:

* Visual Studio 2019
* Microsoft Word
* SDL2
* Online Resources
  1. Drawing circles: https://stackoverflow.com/questions/28346989/drawing-and-filling-a-circle#28351443

# Tasks undertaken:

* I copied the task 24 spike report into the task folder, stripping out the spike report’s original content and replacing it with goals and resources pertaining to the task at hand.
* Using the GameEntity class as a base class, I added the Shape class, which stores a ShapeType and SDL\_Colour. Using Shape as a base class, I added the Circle and Rectangle classes. Circles store a radius value, while Rectangles store a width and height.

# What we found out: