**Title:** Task B-1 Report

**Author:** Robin Findlay-Marks, s103603871

**Goals / deliverables:**

The goal of this spike plan was to create a program that could create 2d entities that have a collision system.

Items created during this task:

* Code see cos30031-103603871\22 - Spike – Collisions
* Task 22.sln
* Collisions Working.png

**Technologies, Tools, and Resources used:**

The tools used to create this game were:

* Microsoft visual studio
* Microsoft Word

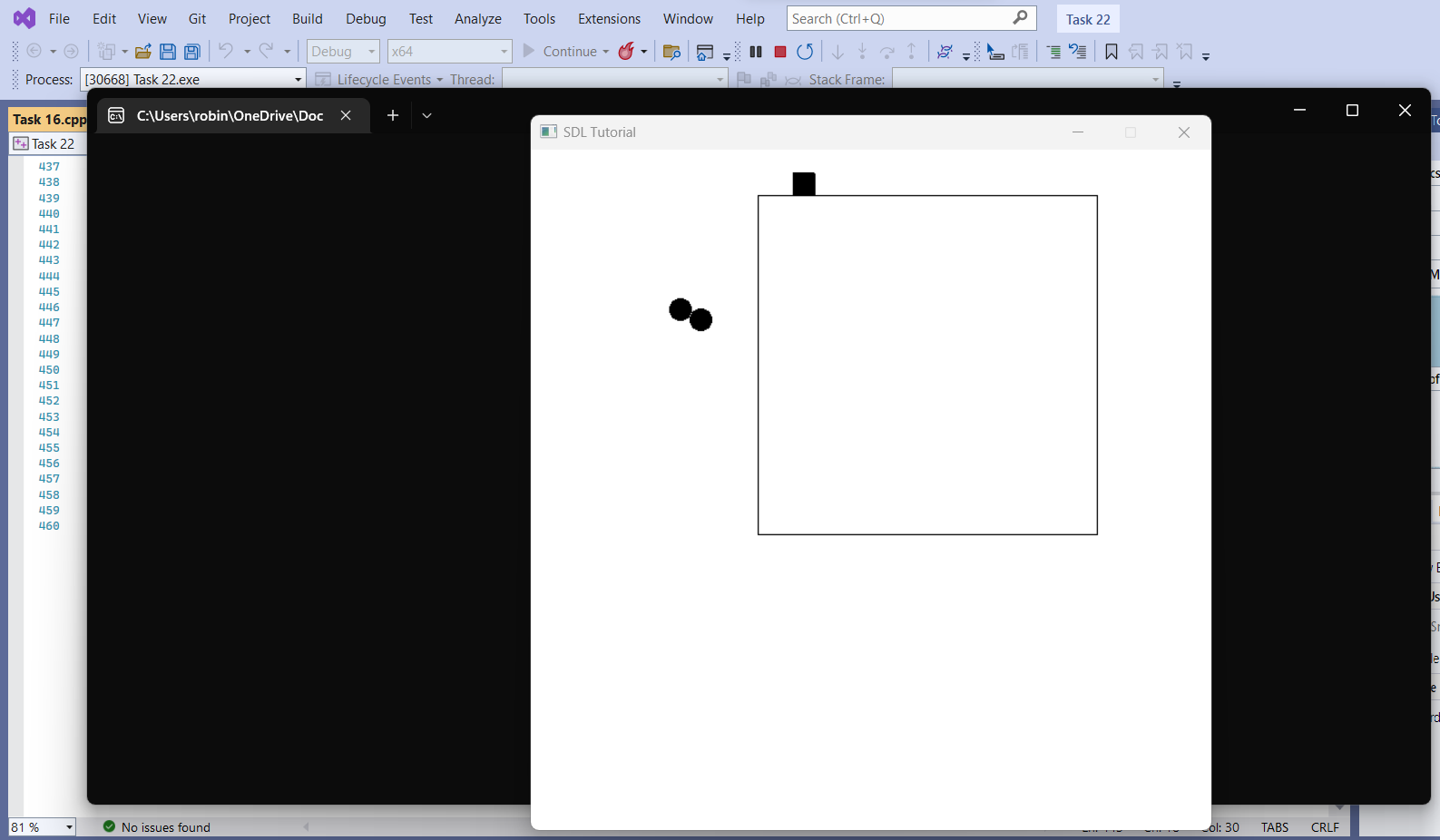
**Tasks undertaken:**

* Follow the lazyfoo tutorial: <https://lazyfoo.net/tutorials/SDL/27_collision_detection/index.php>
* Once the program is working, follow the lazyfoo tutorial: <https://lazyfoo.net/tutorials/SDL/29_circular_collision_detection/index.php>, however this time add the code to the already existing program, rather than making a new one.
* Make sure the two systems run at the same time and using as many of the same functions (minimal duplicate code) as possible.

**What we found out:**

I found this task to be quite easy. There were online tutorials (by lazyfoo productions) I could follow which told me exactly what to do (<https://lazyfoo.net/tutorials/SDL/27_collision_detection/index.php> and <https://lazyfoo.net/tutorials/SDL/29_circular_collision_detection/index.php>). However they did it in two separate programs and I had to merge the two tutorials together into one working system. It took a bit of time but I managed to get it to work. I had an issue where the game run way too fast where pressing an arrow key would cause the square or circle to immidietly move to the edge of the screen. I had to redo some of my code (I never found out exacly what was causing it, I even set the movment velocity to the minimun, which only mildly helped).  
  
In my program the box is controlled with the arrow keys and only has collision with the box and the circle is controlled with the wasd key and only has collision with the other circle.

Example of program working:



Lazy Foo' Productions, 23/01/2022, *Lazy Foo' Productions - Collision Detection*, Lazy Foo' Productions, viewed 16/11/2022, < <https://lazyfoo.net/tutorials/SDL/27_collision_detection/index.php>>.

Lazy Foo' Productions, 23/01/2022, *Lazy Foo' Productions - Circular Collision Detection*, Lazy Foo' Productions, viewed 16/11/2022, < <https://lazyfoo.net/tutorials/SDL/29_circular_collision_detection/index.php> >.