**Spike:** Task 17.P

**Title:** Sprites & Graphics

**Author:** Mitchell Wright, 100595153

**Goals / deliverables:**

To demonstrate an understanding of frameworks and spite/image functionality within SDL2.

Items created during task:

* Code, see: \17 - Spike – Sprites and Graphics\SDL2

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

* Visual Studio 2022
* SourceTree
* GitHub
* SDL2 Development Library
* Lecture 3.2 – Data Structures

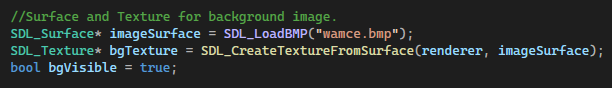
**Tasks undertaken:**

* Toggling Backgrounds
* Using Rectangles to Create Sprite Tiles
* Using Rectangles to Randomise Position
* Commit to Git

**What we found out:**

1. Toggling Backgrounds:

In order to show something within the game window, we need to first create a surface and then use that to create a texture.



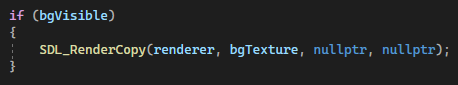
In this case, a bool is also created so that we can toggle the background.



The renderer set up in T15 is cleared at the start of the render loop.



The texture is then copied to the renderer.



And then rendered.



1. Using Rectangles to Create Sprite Tiles:

Rectangles can be used to select portions of textures to render.



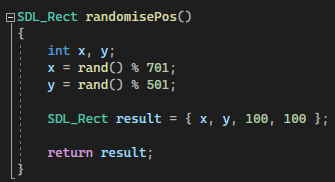
These are used as parameters when copying the texture to the renderer.



And thus only a portion of the original BMP is shown.

1. Using Rectangles to Randomise Position:

In order to randomise the positioning of the rectangle on screen, the following function is used. Nothing too dissimilar to what has been done with randomisation in previous tasks.



This function is called on keypress, same as in the previous task. Once the position has been randomised, the rectangle is passed through to the renderer.



1. Commit to Git:

