**<Sprite Animation>**

For the B grade in class project, I have implemented sprite sheet animation. In my game project, I have implemented the animation already, and I just modified some parts to make it easier to use when designing the game.

The sprite animation constructor is contained in the “Component\_Sprite.cpp” which takes in the 1)Object pointer, 2)sprite path, 3)bool to tell if is animated, 4)number of frames, 5)the animation speed, 6)position, 7)scale. There are a lot of parameters because I wanted the constructor to be as flexible and modifiable as possible. The constructor would then initialize the private members and then load the texture from the given path.

After sending the texture sample to the shader, I create a square mesh, at this point the mesh looks like this :



However, this would look weird in a single square mesh, so we calculate the distance of the first block of the sprite sheet. Since the texture coordinate goes from 0 to 1, we can calculate (1 / number of frames) to get the first block’s offset. Now we only show the the red area :



After the constructor, the Sprite::Update(float dt) function constantly updates the sprite by clearing the previous texture coordinates and then adding the texture coordinates of the next block. If we reach the end of the sprite sheet, we reset it to the first block. This operation repeats and makes the image looks like it is moving seamlessly.