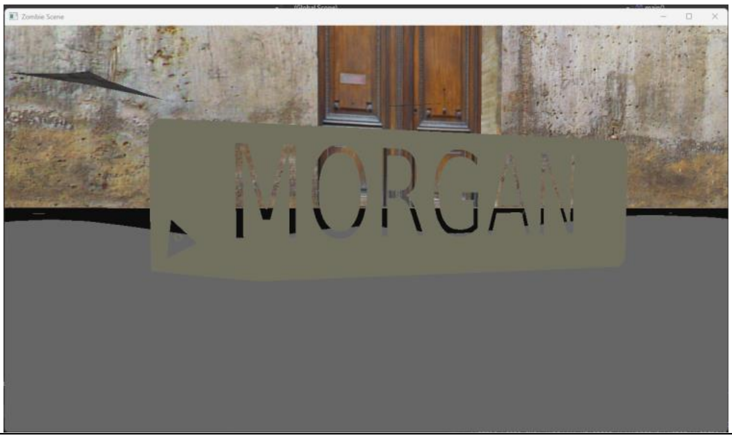
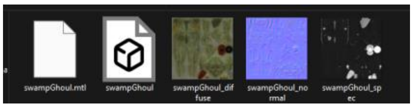
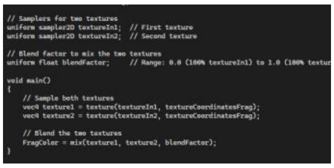
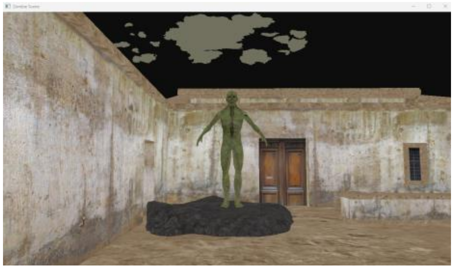
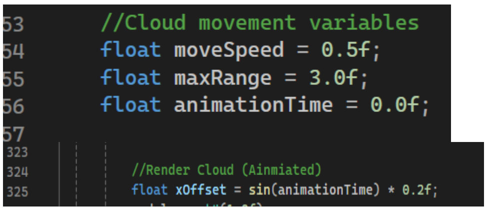
My submission fulfils the following conditions to pass:

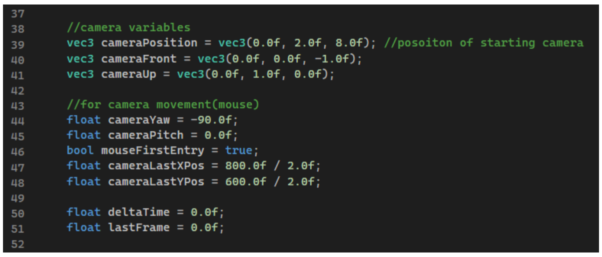
Software compiles	Yes
Populated Git repo consistent with DLE submission exists	Git link: https://github.com/Mdot5596/Zombie-Scene-OpenGL
OpenGL in C++ with vertex and fragment shaders loaded with a quad displayed and signature feature visible.	List your signature(s): My signature is a 3D rectangle with morgan written on it , this is hidden inside of the rock 
Write up and video explanation submitted	Filenames: https://github.com/Mdot5596/Zombie-Scene-OpenGL/blob/main/README.md YouTube: https://youtu.be/z9JmAVT52C4
Defended work in viva.	Yes

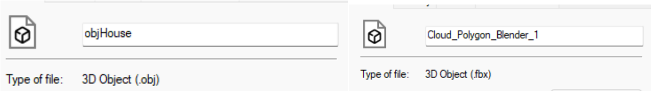
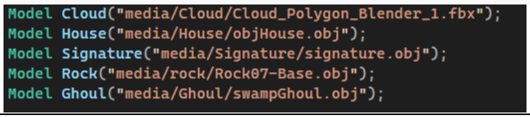
My CW2 project has the following features

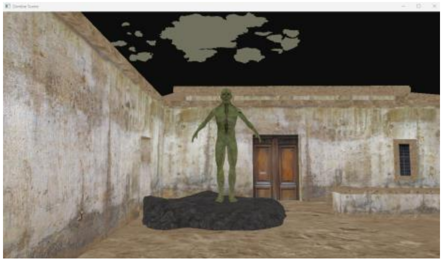
Feature	**Description**	**Category**	**Marks Claimed**	**Marks Awarded (for ML use)**
MVP Implemented	MVP transforms implemented in vertexshader.vert line 12 and 18. And in Main.CPP 33-37, 278-283 ,456-457. MVP done in CPP line 456, which should be done in shaders for better efficiency.	40-70 Marks	5	2.5

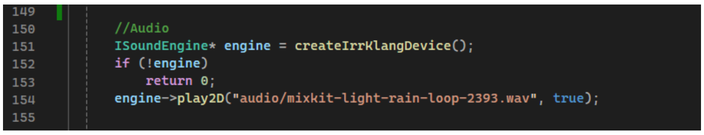
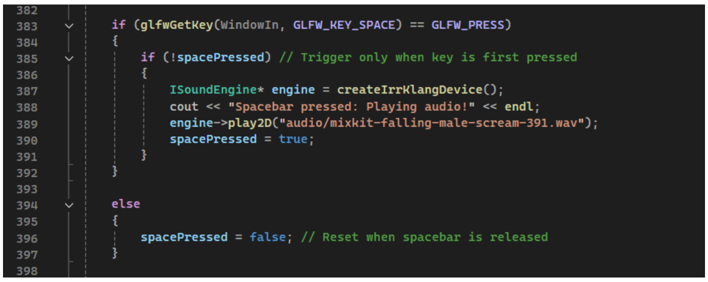
Textures	Single and Mixed textures working in multiple models such as house, rock, and zombie
	Example for Zombie:
	
	 swampGhoul.mtl
	swampGhoul
	swampGhoul_diffuse
	swampGhoul_normal
	swampGhoul_spec
	There you can see the diffuse, normal, and spec textures, and they apply properly to the zombie model:
	
	
	`// Samplers for two textures`
	`uniform sampler2D textureIn1;  // First texture`
	`uniform sampler2D textureIn2;  // Second texture`
	
	`// Blend factor to mix the two textures`
	`uniform float blendFactor;    // Range: 0.0 (100% textureIn1) to 1.0 (100% textur`
	
	`void main()`
	`{`
	`    // Sample both textures`
	`    vec4 texture1 = texture(textureIn1, textureCoordinatesFrag);`
	`    vec4 texture2 = texture(textureIn2, textureCoordinatesFrag);`
	
	`    // Blend the two textures`
	`    FragColor = mix(texture1, texture2, blendFactor);`
	`}`
	Verified mixed textures in fragment shader.
		40-70 Marks	5	5

3D polygons with scene animations	Multiple 3D polygon objects are displayed on the screen – that being the zombie, the house, the rock, the signature, and the clouds
	
	Scene animation being the clouds moving
	See YouTube video for evidence of the clouds being animated
	
	`//Cloud movement variables`
	`float moveSpeed = 0.5f;`
	`float maxRange = 3.0f;`
	`float animationTime = 0.0f;`
	`//Render Cloud (Ainmiated)`
	`float xOffset = sin(animationTime) * 0.2f;`
	Verified cloud has movement timings
		40-70 Marks	5	5

Keyboard/mouse movement	W,A,S,D can be used to fluidly move around the scene and the mouse can be used to look around the screen smoothly.
	
	`//camera variables`
	`vec3 cameraPosition = vec3(0.0f, 2.0f, 8.0f); //posoiton of starting camera`
	`vec3 cameraFront = vec3(0.0f, 0.0f, -1.0f);`
	`vec3 cameraUp = vec3(0.0f, 1.0f, 0.0f);`
	`//for camera movement(mouse)`
	`float cameraYaw = -90.0f;`
	`float cameraPitch = 0.0f;`
	`bool mouseFirstEntry = true;`
	`float cameraLastXPos = 800.0f / 2.0f;`
	`float cameraLastYPos = 600.0f / 2.0f;`
	`float deltaTime = 0.0f;`
	`float lastFrame = 0.0f;`
	See lines 370-453 in Main.CPP for the Process User Input function which holds the movement for keyboard. And see the Mouse callback function which holds the function for mouse movement.
	Verified keyboard and mouse movements work perfectly
		40-70 Marks	5	5

Load models with textures	Multiple Models have been successfully loaded with correct textures, 2 different Model types (.obj and .fbx)
	
	objHouse
	Type of file: 3D Object (.obj)
	Cloud_Polygon_Blender_1
	Type of file: 3D Object (.fbx)
	
	`Model Cloud("media/Cloud/Cloud_Polygon_Blender_1.fbx");`
	`Model House("media/House/objHouse.obj");`
	`Model Signature("media/Signature/signature.obj");`
	`Model Rock("media/rock/Rock07-Base.obj");`
	`Model Ghoul("media/Ghoul/swampGhoul.obj");`
				5

	 Verified obj and fbx models loadable			
Procedural content generation	Successfully generated 3 different biomes using fastnoiselite and Perlin noise to with unique frequencies and amplitude so they can be differentiated. See lines 58 – 142, 197 – 259, 285- 307 I see only one biome? The max mark is for biomes overlapped over a single terrain and clearly visible (as in Lab 8), not 3 separate terrains. Besides, the other 2 are barely noticeable.			2.5

Audio	Added background audio noise (rain) and interactive audio playback (spacebar makes camera scream).
	See YouTube Video for evidence audio works
	
	//Audio
	ISoundEngine* engine = createIrrKlangDevice();
	if (!engine)
	    return 0;
	engine->play2D("audio/mixkit-light-rain-loop-2393.wav", true);
	
	if (glfwGetKey(WindowIn, GLFW_KEY_SPACE) == GLFW_PRESS)
	{
	    if (!spacePressed) // Trigger only when key is first pressed
	    {
	        ISoundEngine* engine = createIrrKlangDevice();
	        cout << "Spacebar pressed: Playing audio!" << endl;
	        engine->play2D("audio/mixkit-falling-male-scream-391.wav");
	        spacePressed = true;
	    }
	}
	
	else
	{
	    spacePressed = false; // Reset when spacebar is released
	}
	
	That’s a pretty hilarious sound effect, but it’s interactive.
		Advanced		5

Research Related Implementation	This has been discussed using 2 examples in my Readme on the Projects Github page I did say check with me about what constitutes proper ‘research’ before applying it. This doesn’t include tutorials.			0
Total Feature Marks			**30**	25
Advanced			**20**	5
Passing Marks			**40**	40
Aesthetics			10	5
Total				75
ML Feedback:				

Nice attempt to include all the important points in the rubric on to the scene. Some were a bit far-fetched but I'll give them to you anyway for the attempt (like the interactive audio). It would be good if the scene can make a bit more sense than a static zombie screaming in the middle of a reck, but I think it's a decent attempt at aesthetics.

Name: Morgan Hodge

*By submitting this form I acknowledge all the information claimed to be true.