Project #3: Interactive Tank

1 Requirements

- Start with `examine` which provieds an interactive UI implementation.
- Render the world space axes as in the video.
- Render the ground grid as in the video.
- Move the tank with arrow keys.
- Rotate the turret with "A" and "D" keys.
- Elevate up/down the barrel with "W" and "S" keys.
- Apply texture mapping as follows.
- Set up the lights appropritely. (In the video only one directional light (sun) is set.)
  - Use `tank-color.jpg` and `shell-color.png` as the diffusive material property.
  - Use the red channel of `tank-etc.png` and `shell-etc.png` as the ambient occlusion value. You can apply this value by multiplying it to the computed diffusive reflection color.
  - Use the green channel of `tank-etc.png` and `shell-etc.png` as the 'roughness' property, which can be assigned to the 'shininess' property for the specular reflection color.
  - Use the blue channel of `tank-etc.png` and `shell-etc.png` as the 'metalness' property, which can be used as the specular reflection material property.
  - Use `tank-normal.png` and `shell-normal.png` as the normal map textures. Refer to `normal_map` for normal perturbation.

2 Reference examples

- `examine`: Interactive UI implementation
- `load_obj`: Loading & rendering a wavefront `*.obj` file
- `load_gltf`: Loading & rendering a GLTF (`*.gltf` or `*.glb`) file
- `xfm_hierarchy_stack`: Hierarchical transformation using a stack
- `normal_map`: Applying normal map texture to a 3d mesh model.

3 Hints & Notes

- `mat4.cameraAim()` might be useful to transform the flying shells.
  - example @webgpufundamentals.org
- For some reason, writing an index buffer of type `Uint16` won't work if the number of indices is odd. (It generates the error message "`Failed to execute 'writeBuffer' on 'GPUQueue': Number of bytes to write must be a multiple of 4`".) So you should convert it to `Uint32` type yourself if this is the case.
- Again, don't forget to press the SUBMIT buttom after uploading your work!!!
- The original tank model can be found at `https://skfb.ly/6BY9t`. The model is modified for easier handling.
- The original shell model can be found at `https://skfb.ly/o9GVX`. It is converted to an `*.obj` format.