//Hao Ai
//haa777

//documentation for assignment4
Steps to finish the question 1 and question 3.
First thing I have to mention is that I deleted all the _uv in the meshes.

In the ColorFragmentShader:
1, changed the vec3 fragmentColor into 2D UV.
2, I also give the new definition of the colour, which I linked it with the myTextSampler and give its colour.

In the TransformVectexShader:
1, changed the vertexNormal_modelspace into vertex_uv.
2, deleted all vec3 ModelColor = vec3(1, 1, 1); vec4 l = M * vec4(vertexNormal_modelspace,0); fragmentColor = ModelColor * max(0,l.x); since they are no more useful.
3, added UV = vertex_uv which I have already gave the definition i the front.

In the part4.cpp
1, I deleted the ProjectionMatrix since they are no more used in the shader.
2, set the loadModels “silo.models” into “default.models” as the requirement said so.
3, add three other model in order to scrape them together later on.
4, define a TextureID to linked these 4 models into myTextureSampler and can be found according to the programID.
5, In the set model part. Add the other 3 model and give them attribution including scale, rotate, and translate
6, in the create vertex attributes, I deleted the vertex_indices, uv_indices, normal_indices since in the shader part, I have already deleted them.
7, still in the create vertex attributes, I add the other 3 models’ attributes as well.
8, after finish created the vertex attributes. I defined the size of these 3 models.
9, next, I associate the models with the shaders in order to bind them later on.
10, I changed the 3D into 2 D
11, unbind them and deleted them one by one.

in the question 2, I follow the step which professor gave the links in the assignment, and I imported the silo.obj as my initial model to reform and past the

silo.bmp to to make the object looks like our professor required. One more important part need to mention is that once I past the image on the object the backside cannot be seen, so I add the shadeless to make the back of the image also have the texture.