0616018 林哲宇

GLSL 細節

1. 照抄 spec 的 createShader 把 vert, frag 檔案載入，並 createProgram

```
56  void shaderInit() {
57      //// TODO: ////
58      //
59      // Hint:
60      // 1. createShader
61      // 2. createProgram
62      GLuint vert = createShader("Shaders/vertexShader.vert", "vertex");
63      GLuint frag = createShader("Shaders/fragmentShader.frag", "fragment");
64      program = createProgram(vert, frag);
65      /*-------------------------*/
66  }
```

2. 建立 VAO, VBO，用一個 VAO 去串三個 VBO

```
68  void bindbufferInit() {
69      //// TODO: ////
70      //
71      //   Hint:
72      // 1. Setup VAO
73      // 2. Setup VBO of vertex positions, normals, and texcoords
74      // https://www.itread01.com/content/1541843428.html
75      glGenVertexArrays(1, &VAO);
76      glBindVertexArray(VAO);
77      glGenBuffers(3, VBO);
78      glBindBuffer(GL_ARRAY_BUFFER, VBO[0]);
79      glBufferData(GL_ARRAY_BUFFER, sizeof(float) * model->positions.size(), &model->positions[0], GL_STATIC_DRAW);
80      glVertexAttribPointer(0, 3, GL_FLOAT, GL_FALSE, 0, (void*)0);
81      glEnableVertexAttribArray(0);
82      glBindBuffer(GL_ARRAY_BUFFER, 0);
83
84      glBindBuffer(GL_ARRAY_BUFFER, VBO[1]);
85      glBufferData(GL_ARRAY_BUFFER, sizeof(float) * model->normals.size(), &model->normals[0], GL_STATIC_DRAW);
86      glVertexAttribPointer(1, 3, GL_FLOAT, GL_FALSE, 0, (void*)0);
87      glEnableVertexAttribArray(1);
88      glBindBuffer(GL_ARRAY_BUFFER, 0);
89
90      glBindBuffer(GL_ARRAY_BUFFER, VBO[2]);
91      glBufferData(GL_ARRAY_BUFFER, sizeof(float) * model->texcoords.size(), &model->texcoords[0], GL_STATIC_DRAW);
92      glVertexAttribPointer(2, 2, GL_FLOAT, GL_FALSE, 0, (void*)0);
93      glEnableVertexAttribArray(2);
94      glBindBuffer(GL_ARRAY_BUFFER, 0);
95      /*-------------------------*/
96  }
```

3. 在 drawUmbreon 時使用 VAO，另外還有 glGetUniformLocation 將 vert, frag 檔案的內容可以接起來產生出那隻神奇寶貝的樣貌。

```
268  void DrawUmbreon()
269  {
270      glm::mat4 M(1.0f);
271      M = glm::rotate(M, glm::radians(angle), glm::vec3(0, 1, 0));
272      M = glm::translate(M, glm::vec3(0, 1.3, 0));
273
274      //// TODO: ////
275      // pass projection matrix, and view matrix and trigger by Uniform (use getP() amd getV())
276      // also pass modeltexture to shader and trigger by Uniform
277      glUseProgram(program);
278      GLuint ModelMatrixID = glGetUniformLocation(program, "ModelView");
279      glUniformMatrix4fv(ModelMatrixID, 1, GL_FALSE, &M[0][0]);
280      GLint pmatLoc = glGetUniformLocation(program, "Projection");
281      GLint mmatLoc = glGetUniformLocation(program, "Vertex");
282      glUniformMatrix4fv(pmatLoc, 1, GL_FALSE, &getP()[0][0]);
283      glUniformMatrix4fv(mmatLoc, 1, GL_FALSE, &getV()[0][0]);
284      glActiveTexture(GL_TEXTURE0);
285      glBindTexture(GL_TEXTURE_2D, modeltexture);
286      GLint texLoc = glGetUniformLocation(program, "Texture");
287      glUniform1i(texLoc, 0);
288      /*-------------------------*/
289      glBindVertexArray(VAO);
290      glDrawArrays(GL_QUADS, 0, 40 * model->fNum);
291
292      glBindVertexArray(0);
293      glActiveTexture(0);
294      glUseProgram(0);
295  }
```

遇到的問題

很多東西都還不熟，突然要寫一個這樣的東西有點不知所措，最後是去網路上的教學從基礎開始一點一點拼起來。希望 spec 在教的時候不要一個一個 function 分開教，而是可以開很多小 script 分別做到一些事情，然後才要求我們做出成品。不然給一堆散的 function 有些也不知道要填什麼，給了也等於沒給。

Bonus

我做了兩個，一個是按按鍵換 basis 的 texture

```
163    void keyboard(unsigned char key, int x, int y) {
164        switch (key)
165        {
166        case 'w':
167            LoadTexture(basistexture, "basis.jpg");
168            break;
169        case 's':
170            LoadTexture(basistexture, "love.jpg");
171            break;
172        }
173    }
```

另一個是改 frag 檔案改神奇寶貝的顏色，讓它顏色變淺

```
1   #version 430
2   //// TODO ////
3   //
4   // Hint:
5   // 1. Recieve texcoord and Normal from vertex shader
6   // 2. Calculate and return final color to opengl
7   //
8   uniform sampler2D Texture;
9
10  in vec2 frag_texcoord;
11
12  out vec4 color;
13  void main(){
14      // bonus: color * 4
15      color = 4 * texture2D(Texture, frag_texcoord);
16  }
17
```