

CG_HW1_Report

操作說明:

h – to show help menu

z/x -switch different models(there are 6 models)

w – switch solid/wireframe mode

c –color filter function

實作:

Normalize:

首先我們要將頂點的 x,y,z 分別存進 vertices 中，以每三個一組

```
vertices[i * 9 + 0] = vx;
//printf("fuck2");
vertices[i * 9 + 1] = vy;
vertices[i * 9 + 2] = vz;

vertices[i * 9 + 3] = vx;
vertices[i * 9 + 4] = vy;
vertices[i * 9 + 5] = vz;

vertices[i * 9 + 6] = vx;
vertices[i * 9 + 7] = vy;
vertices[i * 9 + 8] = vz;
```

對於顏色部分也是將 r,g,b 分開存進 color 裡，一樣也是三個一組

```
c1 = OBJ->colors[indv1*3+0];
c2 = OBJ->colors[indv1*3+1];
c3 = OBJ->colors[indv1*3+2];
colors[i * 9 + 0] = c1;
colors[i * 9 + 1] = c2;
colors[i * 9 + 2] = c3;
```

```
c1 = OBJ->colors[indv2*3+0];
c2 = OBJ->colors[indv2*3+1];
c3 = OBJ->colors[indv2*3+2];
colors[i * 9 + 3] = c1;
colors[i * 9 + 4] = c2;
colors[i * 9 + 5] = c3;
c1 = OBJ->colors[indv3*3+0];
c2 = OBJ->colors[indv3*3+1];
c3 = OBJ->colors[indv3*3+2];
colors[i * 9 + 6] = c1;
colors[i * 9 + 7] = c2;
colors[i * 9 + 8] = c3;
```

然後開始做 translation 的前置步驟，關於這部分我是先利用助教寫好的 max 跟 min function 將圖型頂點中的 x,y,z 軸各自最小最大值分別存進 minVal、maxVal 之中 maxVal[0]=x 軸最大值，以此類推。做好這步驟後就可以開始對每個點做 translation，做法是如下

```

maxVal[0] = max(maxVal[0], vx);
minVal[0] = min(minVal[0], vx);
maxVal[1] = max(maxVal[1], vy);
minVal[1] = min(minVal[1], vy);
maxVal[2] = max(maxVal[2], vz);
minVal[2] = min(minVal[2], vz);

```

之後開始做 translation，對於每個我們分好的點，也就是 x,y,z 分別去減去 (maxVal+minVal)/2，也就是達到平移到物體的中心，code 如下。

```

for (i = 0; i < 3 * (int)OBJ->numtriangles; i++) {
    vertices[i * 3 + 0] -= (maxVal[0] + minVal[0]) / 2;
    vertices[i * 3 + 1] -= (maxVal[1] + minVal[1]) / 2;
    vertices[i * 3 + 2] -= (maxVal[2] + minVal[2]) / 2;
}

```

接著做 scaling，先找出 x,y,z 中 maxVal-minVal 的值當作要除掉的值，這樣的做法在於避免圖型因為除以不同的差值產生變形，之後對於每個點除以剛剛找出的值 scale 在乘上 2=1-(-1)，因為是 normalize 到[-1 1]，所以還要乘上，code 如下。

```

float scale = max((maxVal[0] - minVal[0]), (maxVal[1] - minVal[1]));
scale = max(scale, (maxVal[2] - minVal[2]));

for (i = 0; i < 9 * (int)OBJ->numtriangles; i++) {
    vertices[i] = (2 * vertices[i] / scale);
    //printf("%d ", vertices[i]);
}

```

Menu:

```

case 'h':
    printf("\n%s\n -to show help menu\n/nz/x -switch different models\nw -switch solid/wireframe mode\nc -color filter function\n%s\n",line,line);
    break;

```

Change model:

將 filename 改成 2 階 char 陣列，然後在 loadmodel 改成 filename[index]，也就是目前要讀的 model，在切換的部分對於每次轉換去呼叫 loadmodel()即可。

```

char filename[][200] OBJ = glmReadOBJ(filename[index]);

case 'z':
    index = (index - 1) % 6;
    if (index < 0) index = 5;
    loadOBJModel();
    //printf("%d", index);
    break;
case 'x':
    index = (index + 1) % 6;
    loadOBJModel();
    //printf("%d", index);
    break;

```

Solid/wireframe:設一個變數 `polygon` 去控制要畫實心還是單純結構，利用 `glPolygonMode` 去實作，solid 是 `GL_FILL`，wireframe 則是 `GL_LINE`，一樣要去呼叫 `loadmodel()`。

```
case 'w':
    polygon =(polygon+1)%2;
    loadOBJModel();
    break;
if (polygon) {
    glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);
}
else {
    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
}
```

Color filter:

一開始先利用 `glGetUniformLocation` 跟 shader 中的 `colorflag` 取得連結

```
color_link = glGetUniformLocation(p, "colorflag");
```

之後在按下 c 時將目前的 `colormode` 傳給 `colorflag` 去做判斷，透過 `glUniform1i` 將 `int colormode` 傳進去

```
case 'c':
    colormode = (colormode + 1) % 4;
    //setShaders();
    //loadOBJModel();
    printf("%d\n", colormode);
    glUniform1i(color_link,colormode);
    break;
```

Shader.frag 如下，加入新 `uniform int` 變數 `colorflag`，然後依據 `colorflag` 去對色彩進行屏蔽。

```
uniform int colorflag;
void main() {
    if(colorflag==1){
        gl_FragColor=vec4(vv3color.r,0,0,1.0);
    }
    else if(colorflag==2){
        gl_FragColor=vec4(0,vv3color.g,0,1.0);
    }
    else if(colorflag==3){
        gl_FragColor=vec4(0,0,vv3color.b,1.0);
    }
    else {
        gl_FragColor = vec4(vv3color, 1.0);
    }
}
```

遇到的問題:

一開始遇到的問題是忘記去 `new float` 給 `vertices` 跟 `colors`，除此外沒太大的問題。

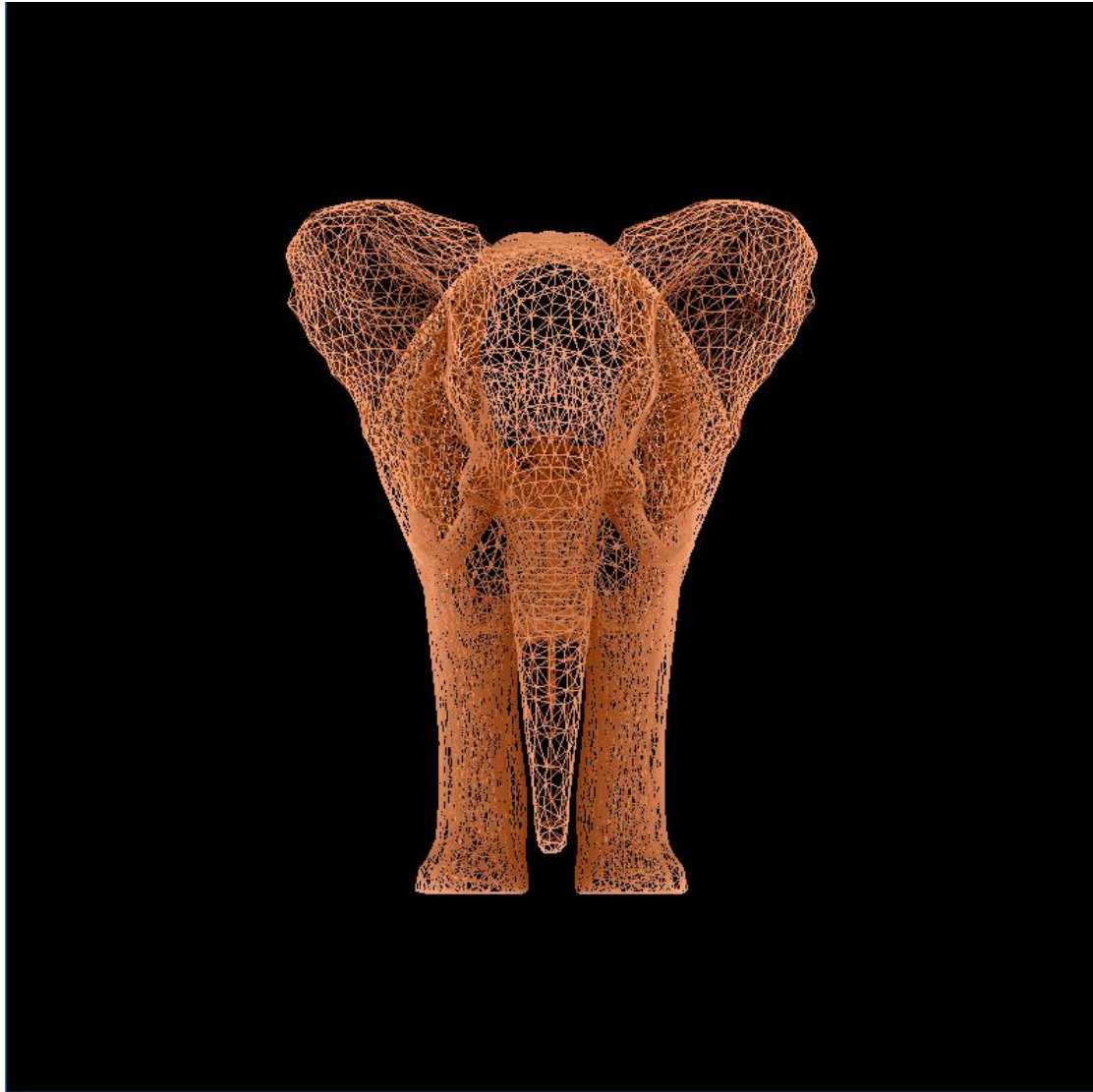
螢幕截圖:

Origin:



W-:

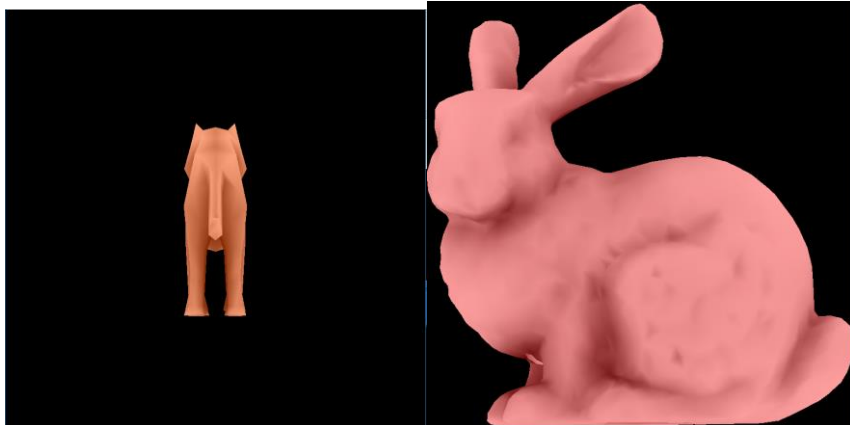




h-:

```
h -to show help menu  
z/x -switch different models  
w -switch solid/wireframe mode  
c -color filter function
```

z/x:



C:

