

Assignment #2

实现正交投影：

1. 在 Camera.h 中，声明函数 setOrthographicParams() 以及参数 m_near、m_far、m_width 和 m_height。其中，m_near 为近平面距离，m_far 为远平面距离，m_width 为平面的宽，m_height 为平面的高。具体代码如下：

```
public:  
    void setOrthographicParams(f32 near, f32 far, f32 width, f32 height);
```

```
protected:  
    f32 m_near, m_far;  
    f32 m_width, m_height;
```

2. 在 Camera.cpp 中，实现该函数。其中，在函数体内，四个参数会传入函数 OrthographicLHMatrix() 中，得到的返回值就是投影矩阵。具体代码如下：

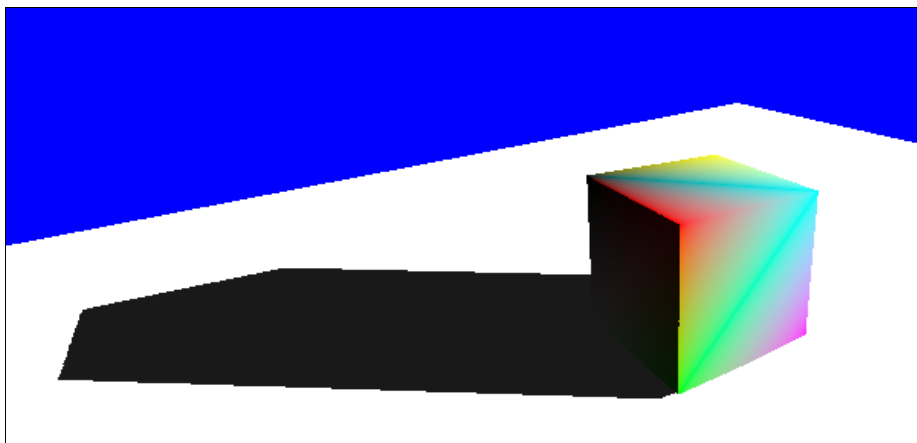
```
void Camera::setOrthographicParams(f32 near, f32 far, f32 width, f32 height)  
{  
    m_near = near;  
    m_far = far;  
    m_width = width;  
    m_height = height;  
  
    m_projMat = Matrix4f::OrthographicLHMatrix(m_near, m_far, m_width, m_height);  
}
```

3. 在 ShadowMapApp.cpp 中，将透视投影函数的调用注释掉，然后调用正交投影函数，并给参数赋值。经过多次尝试，当 near 为 1.0f，far 为 50.0f，width 为 10.0f，height 为 10.0f 时，效果较好。具体代码如下：

```
//m_camLight.setProjectionParams(0.25f * Pi, AspectRatio(), 1.0f, 50.0f);  
m_camLight.setOrthographicParams(1.0f, 50.0f, 10.0f, 10.0f);
```

最终效果：

下图为侧视效果：



下图为俯视效果：

