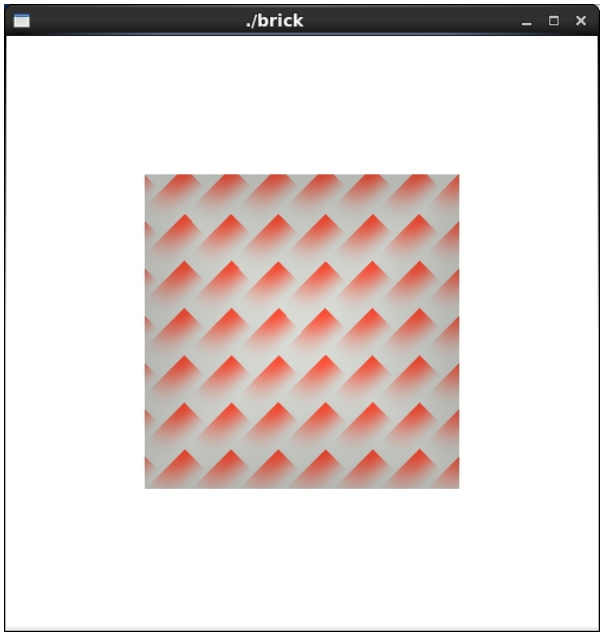
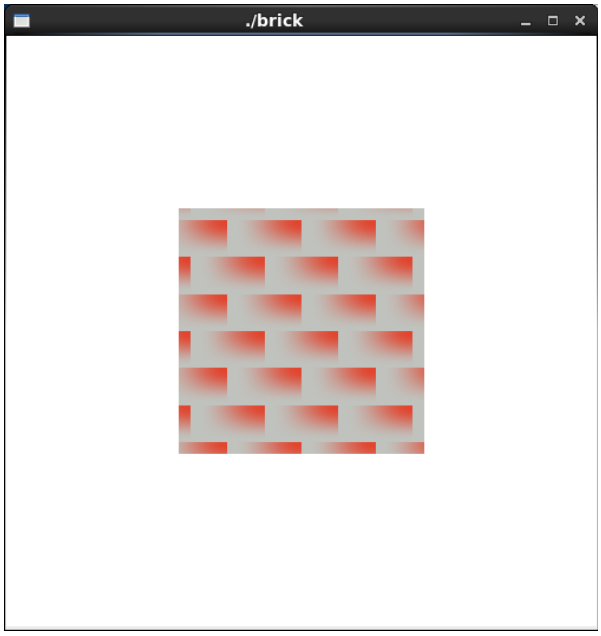
Yazhuo Liu
Lab 10

Write a shader program that displays a brick wall constructed using triangluar bricks.





Code:

//brick.cpp

```
...
void display(void)
{
  GLfloat vec[4];

  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
  glClearColor( 1.0, 1.0, 1.0, 0.0 );              //get white background color

  glPushMatrix();
  glRotatef( anglex, 1.0, 0.0, 0.0);               //rotate the cube along x-axis
  glRotatef( angley, 0.0, 1.0, 0.0);               //rotate along y-axis
  glRotatef( anglez, 0.0, 0.0, 1.0);               //rotate along z-axis

  GLUquadric *qob = gluNewQuadric();
  gluQuadricTexture(qob, GL_TRUE);

  if ( objectType == 0 )
    glutSolidCube(1.0);
  else {
    glRotatef( 45, 0, 0, 1 );
    gluDisk(qob, 0, 1, 4, 10);
  }
```

```
   gluDeleteQuadric(qob);

   glPopMatrix();
   glutSwapBuffers();
   glFlush();
}
...

//brick.frag

...
void main(void)
{
   vec3  color;
   vec2  position, useBrick;

   position = MCposition / BrickSize ;

   if (fract(position.y * 0.5) > 0.5)
      position.x += 0.5;

   position = fract(position);

   useBrick = smoothstep(position + 0.8, position - 0.1, BrickPct);

   color  = mix(MortarColor, BrickColor, useBrick.x * useBrick.y);
   color *= LightIntensity;
   gl_FragColor = vec4 (color, 1.0);
}
```

Report:

I have tried lots of approaches to modify the step() function so it draws another mortarColor line inside each brick, but none of them worked. I feel like I'm really close to the answer but there's always something not correct. At the end, I've got nothing else I could do so I had to use the smoothstep() function to make the bricks look a little bit like triangles. Therefore, I'm deducting 5 points from the total points.