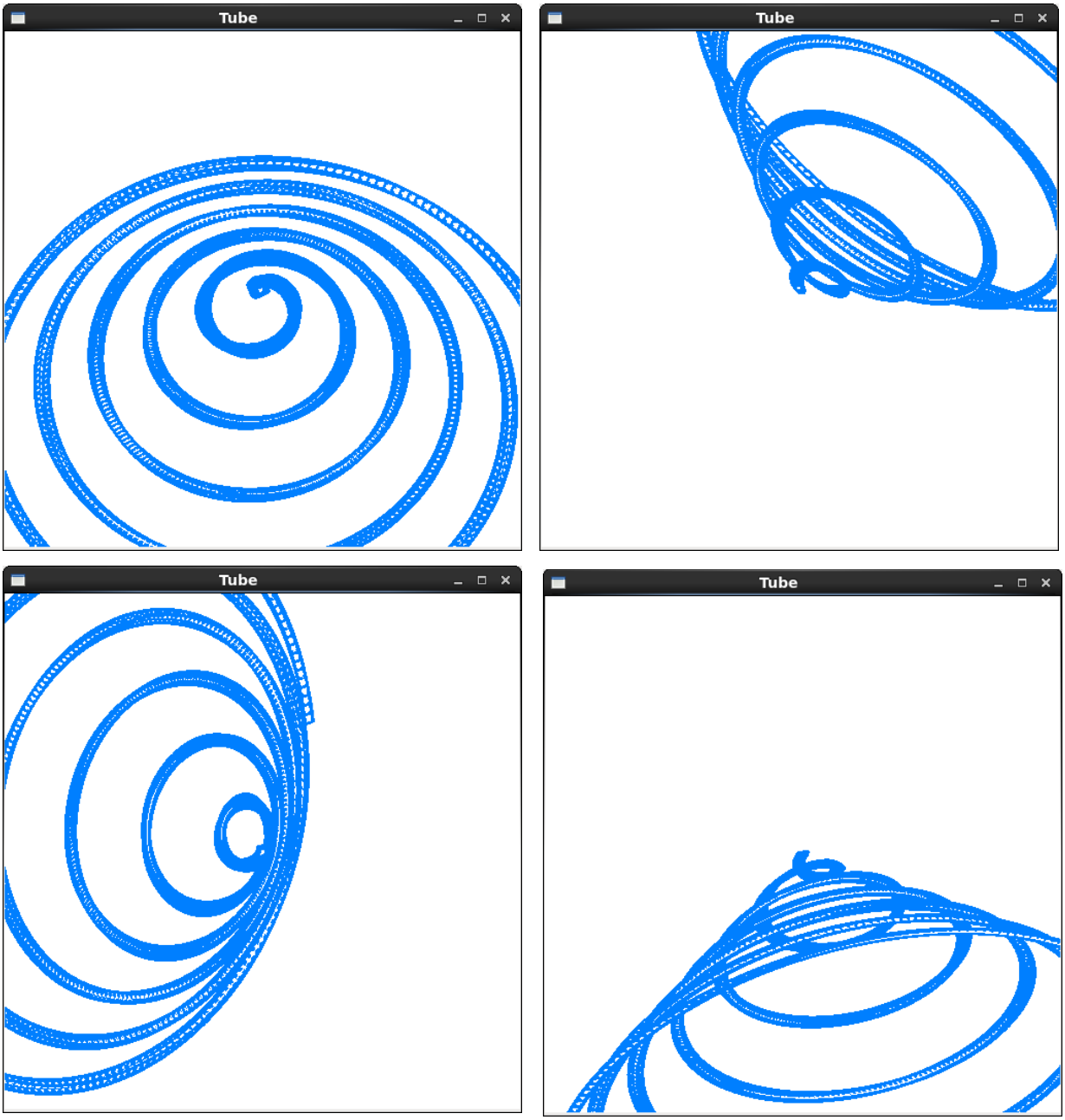
Yazhuo Liu
Lab 15

***Use the extrusion method discussed in class to render a conical helix tube.***



Code:

//conical.cpp

```
...
//helix curve
void get_C ( float C[4], float t, float b )
{
  C[0] = t * cos ( b * t );
  C[1] = t * sin ( b * t );
  C[2] = t;
  C[3] = 1;
}

//Matrix for transforming to Frenet frame
void setM( LinearMapR4 &M, float t, float b )
{
  float c = 1.0 / sqrt ( 1 + b*b );
  M.SetColumn1( -cos(t), -sin(t), 0, 0 );                 //Normal    N(t)
  M.SetColumn2( sin(t)*b*c, -cos(t)*b*c, c, 0 );      //Binormal  B(t)
  M.SetColumn3( -sin(t)*c, cos(t) *c, b * c, 0);        //Tangent   T(t)
  M.SetColumn4( t*cos(b*t), t*sin(b*t), t, 1 );                    //The curve C(t)
}

...

void display(void)
{
  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

  const float b = 6;                //constant of Helix curve
  double H = 6.0;
  LinearMapR4 M;                //Transformation matrix
  const int N = 4;                //number of vertices in base

  vector<Cfloat3>vp0(N), vp1(N);
  VectorR4 p_1;                              //transformed point

  VectorR4 points[4];            //define four points
  points[0] =  VectorR4 (-0.1, -0.1, 0, 1 );     //x, y, z, w
  points[1] =  VectorR4 ( 0.1, -0.1, 0, 1 );     //x, y, z, w
  points[2] =  VectorR4 ( 0.1,  0.1, 0, 1 );     //x, y, z, w
  points[3] =  VectorR4 ( -0.1, 0.1, 0, 1 );     //x, y, z, w

  glColor3f ( 1, 1, 1 );
  glPushMatrix();
  glRotatef( anglex, 1.0, 0.0, 0.0);   //rotate the object about x-axis
  glRotatef( angley, 0.0, 1.0, 0.0);   //rotate about y-axis
  glRotatef( anglez, 0.0, 0.0, 1.0);   //rotate about z-axis
```

```
  float C[4];
  glLineWidth ( 3 );
  glPolygonMode( GL_FRONT, GL_LINE );
  glPolygonMode( GL_BACK, GL_LINE );

  //The curve
  glBegin(GL_LINE_STRIP );
  for ( float t = 0; t <= 2*3.1415926; t += 0.01 ) {
    get_C ( C, t, b );
     glVertex4fv( C );
  }
  glColor3f ( 0, 0.5, 1 );
  glEnd();

  float p3[3];              //3-D point, (x, y, z)
  //starting
  setM ( M, 0, b );       //t = 0
  for ( int i = 0; i < 4; ++i ) {
    p_1 = M * points[i];          //transform the point
    p_1.Dump( vp0[i].p3 );      //put (x, y, z) in vp0[i].p3[]
  }
  glBegin( GL_QUADS );                //a side has four points
  for ( float t = 0.01; t <= 2*3.1415926; t += 0.005 ) {
    setM ( M, t, b );
    for ( int i = 0; i < N; ++i ) {
      p_1 = M * points[i];        //transform the point
      p_1.Dump( vp1[i].p3 );    //put (x, y, z) in vp1[i].p3[]
    }
    for ( int i = 0; i < N; ++i ) { //draw the N sides of tube between 'base' and 'cap'
      int j = (i+1) % N;
      glVertex3fv( vp0[i].p3 );
      glVertex3fv( vp0[j].p3 );
      glVertex3fv( vp1[j].p3 );
      glVertex3fv( vp1[i].p3 );
    }
    copy ( vp1.begin(), vp1.end(), vp0.begin() );        //copy vp1 to vp0
  }  //for t
  glEnd();
  glPopMatrix();
  glFlush();
}
```

Report:

I replaced the formulas in get_C() and setM() to get a new spiral tube, and then I changed the value of b and t in display(). b is a constant which determines the height of the conical helix, and t is the variable goes from 0 to 2 pi.

I have successfully completed the lab.