Министерство Образования и Науки РФ

Санкт-Петербургский государственный электротехнический университет “ЛЭТИ” им. В.И. Ульянова (Ленина)

ОТЧЕТ

по лабораторно-практической работе №3

по дисциплине «Компьютерная графика»

Выполнил: Козей А. В.

Факультет: КТИ

Группа №: 3381

Преподаватель: Герасимова Т.В.

Выполнено 25. 10. 2015г.

Подпись преподавателя \_\_\_\_\_\_\_\_\_

**Цель работы:** Реализовать интерактивное приложение, отображающее B-сплайн(n=6, k = 3), позволяющая пользователю: интерактивно менять положение контрольных точек, касательных, натяжений.

**Ход работы**

1. **B-spline.**

Расчёты:

public class BSplineV2

{

private const int PART\_SIZE = 4;

public class FloatPoint

{

public FloatPoint() { }

public FloatPoint(Point p)

{

X = (float)p.X;

Y = (float)p.Y;

}

public FloatPoint(float x, float y)

{

X = x;

Y = y;

}

public FloatPoint Copy()

{

return new FloatPoint(X, Y);

}

public float X { get; set; }

public float Y { get; set; }

}

public List<FloatPoint> GenerateSpline(List<Point> points)

{

List<FloatPoint> spline = new List<FloatPoint>();

for(int i = 0; i < ((points.Count - PART\_SIZE) + 1); i++)

{

Point p1 = points[i];

Point p2 = points[i + 1];

Point p3 = points[i + 2];

Point p4 = points[i + 3];

float[] a = new float[4];

float[] b = new float[4];

a[0] = (-p1.X + 3 \* p2.X - 3 \* p3.X + p4.X) / 6.0f;

a[1] = (3 \* p1.X - 6 \* p2.X + 3 \* p3.X) / 6.0f;

a[2] = (-3 \* p1.X + 3 \* p3.X) / 6.0f;

a[3] = (p1.X + 4 \* p2.X + p3.X) / 6.0f;

b[0] = (-p1.Y + 3 \* p2.Y - 3 \* p3.Y + p4.Y) / 6.0f;

b[1] = (3 \* p1.Y - 6 \* p2.Y + 3 \* p3.Y) / 6.0f;

b[2] = (-3 \* p1.Y + 3 \* p3.Y) / 6.0f;

b[3] = (p1.Y + 4 \* p2.Y + p3.Y) / 6.0f;

//spline.Add(new FloatPoint(a[3], b[3]));

float point\_count = 1000;// (float)Math.Sqrt(Math.Pow(p2.X - p1.X, 2F) + Math.Pow(p2.Y - p1.Y, 2F));

for(int j = 1; j < point\_count; j++)

{

float t = j / point\_count;

FloatPoint p = new FloatPoint();

p.X = (a[2] + t \* (a[1] + t \* a[0])) \* t + a[3];

p.Y = (b[2] + t \* (b[1] + t \* b[0])) \* t + b[3];

spline.Add(p);

}

}

spline.Insert(0, new FloatPoint(points.First()));

spline.Add(new FloatPoint(points.Last()));

return spline;

}

}

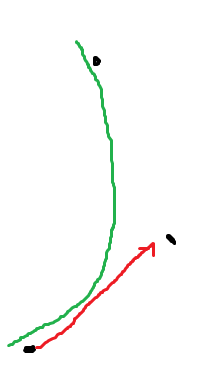
1. Касательная

Черные точки – каркас сплайна

Зелёная кривая – часть сплайна

Красная прямая – касательная

Как-раз таки угол этой касателной мы и будем изменять.



public class Tangent

{

private float k = 0.0f, off = 0.0f;

private float alpha = 0.0f;

public Point Start, End;

public Tangent(Point a, Point b)

{

try

{

k = (b.Y - a.Y) / (b.X - a.X);

off = b.Y - b.X \* k;

}

catch(Exception ex)

{

k = 0; off = 0;

}

Start = new Point(a.X, a.Y);

End = new Point(b.X, b.Y);

}

public void Rotate(float alpha)

{

float x = End.X;

float y = End.Y;

float x0 = Start.X;

float y0 = Start.Y;

End.X = (int)(x0 + (x - x0) \* Math.Cos(alpha) - (y - y0) \* Math.Sin(alpha));

End.Y = (int)(y0 + (y - y0) \* Math.Cos(alpha) + (x - x0) \* Math.Sin(alpha));

}

}

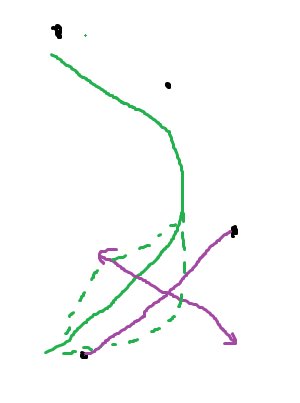
1. Натяжение

Черные точки – точки каркаса сплайна

Зелёная кривая - сплайн

Фиолетовые стрелочки - направление натяжения

Зелёная прерывистая кривая - результат натяжения в ту или иную сторону



public class Tension

{

public Point Left { get; set; }

public Point Right { get; set; }

public Point Center { get; set; }

private float gk = 0.0f;

private float goff = 0.0f;

public Tension(Point left, Point center, Point right)

{

Left = left;

Right = right;

Center = center;

float k = (float)(Left.Y - Right.Y) / (float)(Left.X - Right.X);

float off = Left.Y - Left.X \* k;

gk = -1 / k;

goff = Center.Y - gk \* Center.X;

}

public float Sqr(float a)

{

return a \* a;

}

public void Tense(int val)

{

float a = Center.X;

float c = gk \* a + goff;

float d = val;

float k = gk;

float b = goff;

float new\_x = (float)((2 \* a - 2 \* b \* k + 2 \* c \* k) + Math.Sign(val) \* Math.Sqrt(Sqr(-2 \* a + 2 \* b \* k - 2 \* c \* k) - 4 \* (k \* k + 1) \* (a \* a + b \* b - 2 \* b \* c + c \* c - d \* d))) / (2 \* (k \* k + 1));

float new\_y = gk \* new\_x + goff;

if(val != 0)

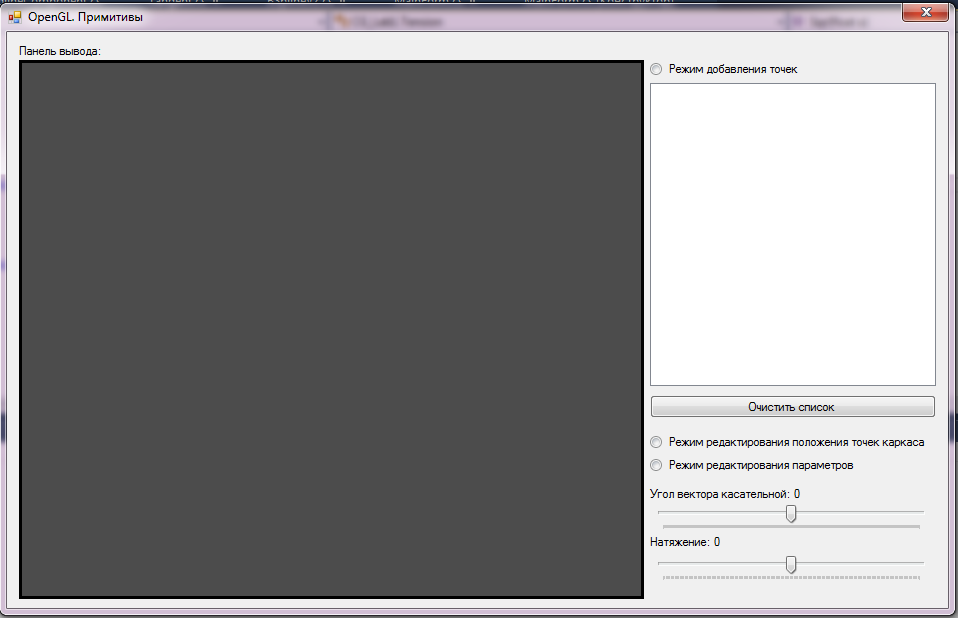
Center = new Point((int)new\_x, (int)new\_y);

}

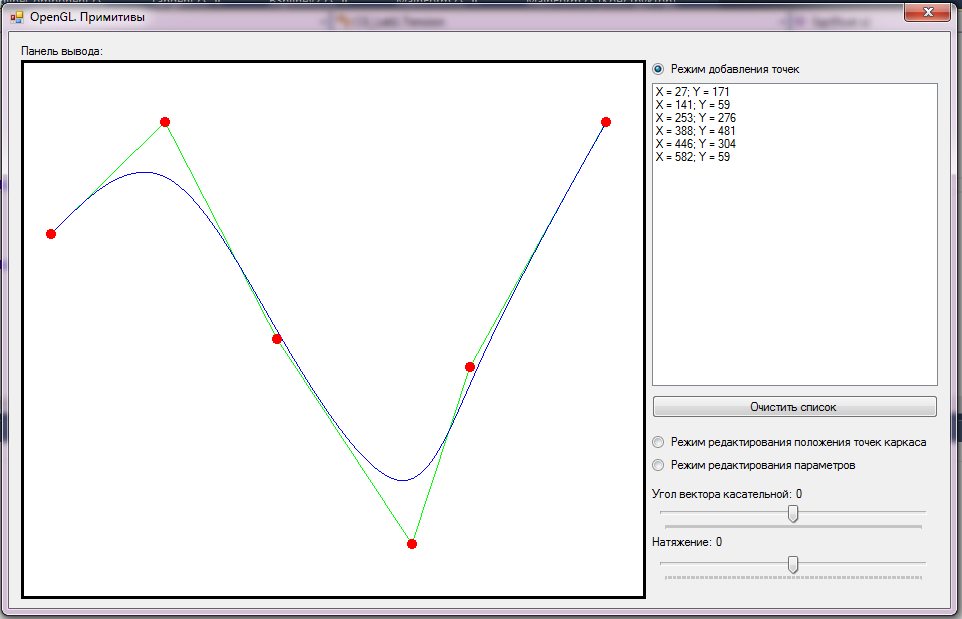
}

**Тестирование**

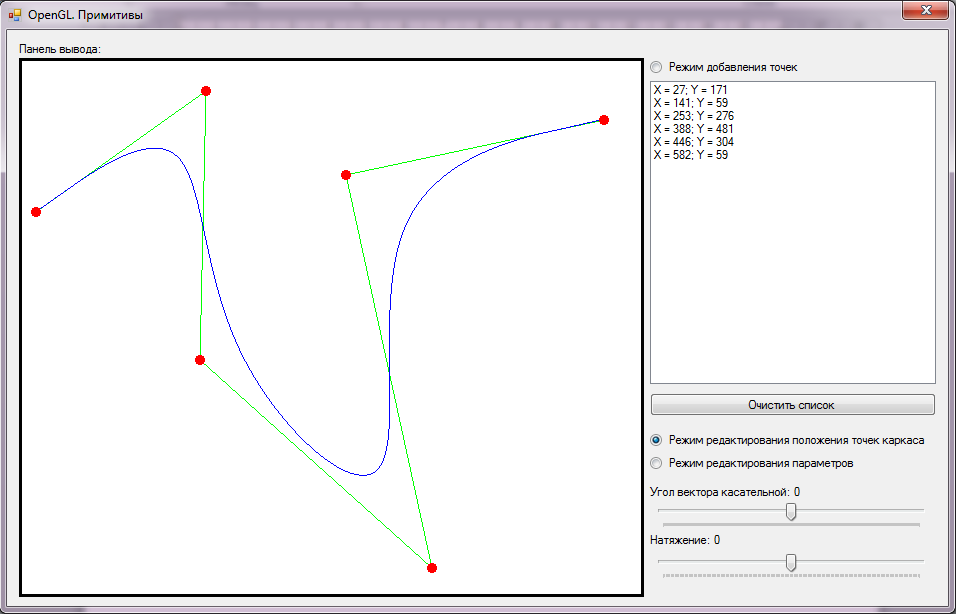
1.Запуск программы



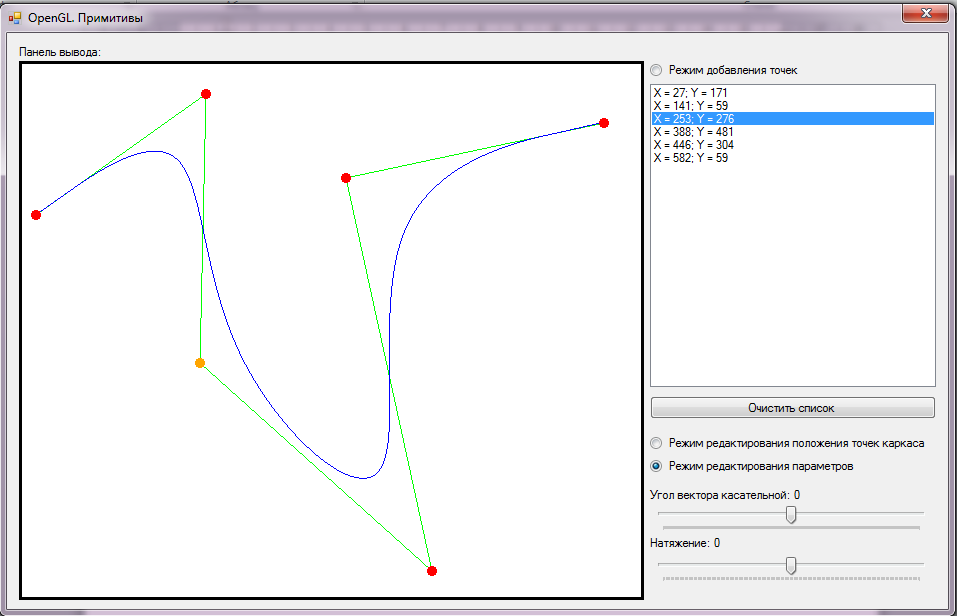
2.Выбираем режим добавления точек и добавляем 6 точек



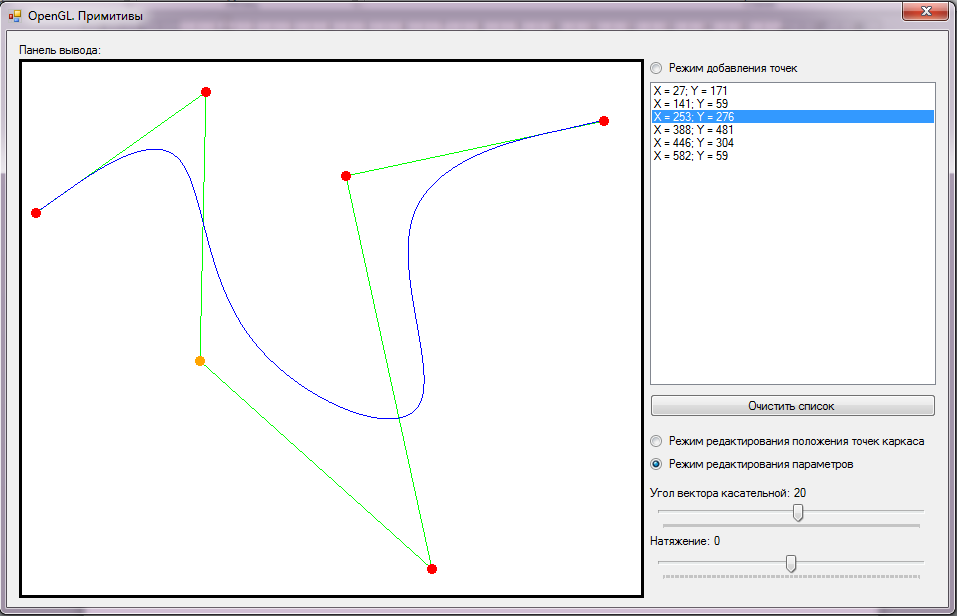
3.Переходим в режим редактирования положения точек каркаса и редактируем каркас, в итоге получится следующее



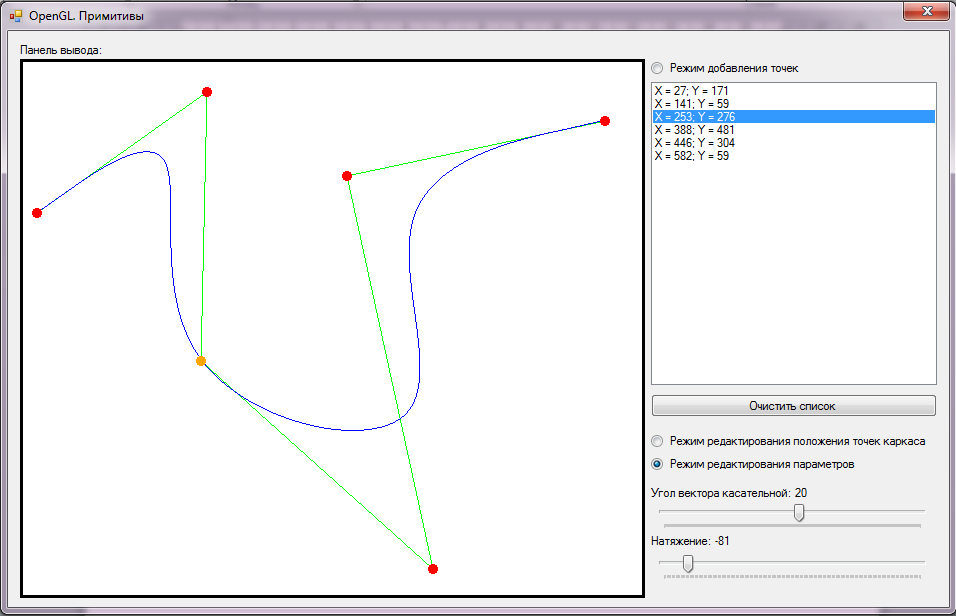
1. Перейдем в режим редактирования параметров и выберем точку(точка отобразится другим цветов:



1. Отредактируем угол вектора касательной:



6.Отредактируем натяжение данной точки:



Вывод: в результате выполнения данной работы я научился работать со сплайнами.