Assignment 5: Bézier Curve

教学班级	专业（方向）	学号	姓名
2班	计算机科学与技术	21307174	刘俊杰

一、基础任务

根据Bézier曲线的定义来计算给定t值（t∈[0,1] ）下的Bézier曲线上的点。

实验原理

Bézier曲线本质上是由调和函数根据控制点插值生成，其参数方程如下：

$$Q(t)=\sum_{i=0}^{n}P_iB_{i,n}(t),t\in[0,1] \tag{1}$$

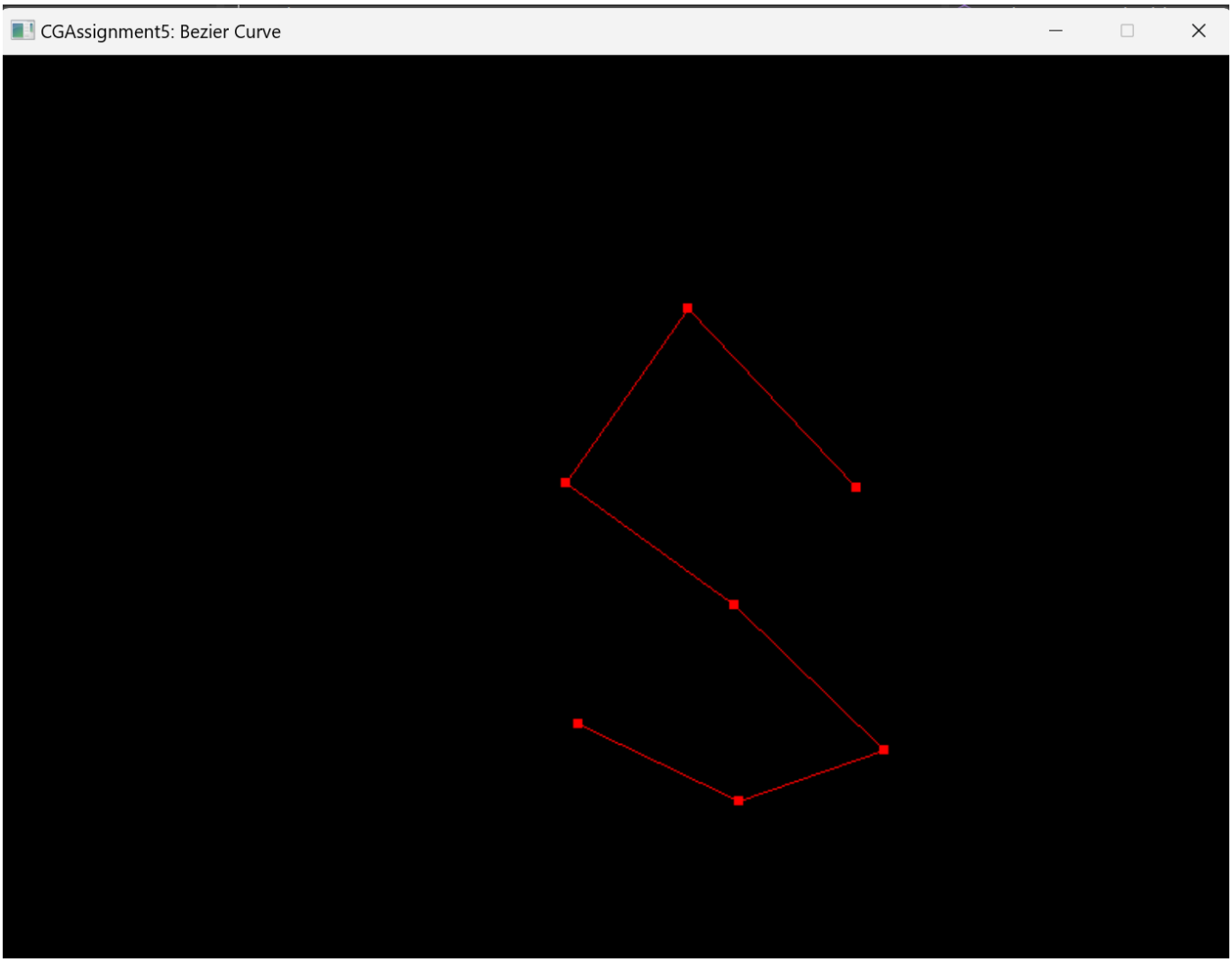
其中Bézier曲线的控制点记为$P_i,i=0,\ldots,n$，一共有n+1个控制顶点。上式是一个n次多项式，一共n+1项，每个控制点对应一项。多项式系数$B_{i,n}$是Bernstein基函数，其数学公式为：

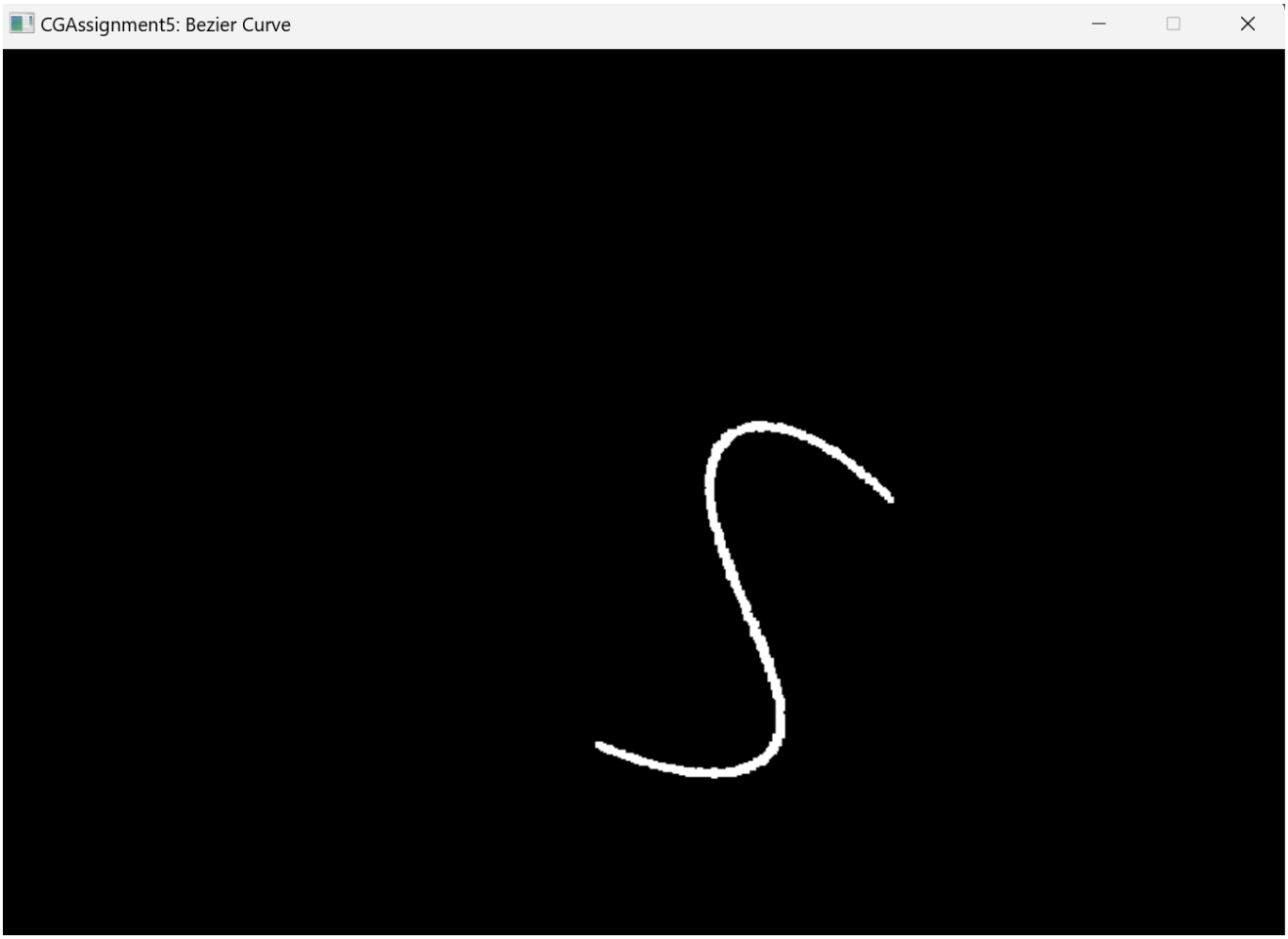
$$B_{i,n}(t)=\frac{n!}{i!(n-i)!}t^i(1-t)^{n-i},i=0,1,\ldots,n \tag{2}$$

代码实现

```
Point2D BezierCurve::basicTask(const std::vector<Point2D> &points, const double
&t) const
{
    //Basic-Task: implement Bezier curve generation algorithm accroding to the
definition
    int n = points.size() - 1;
    float B = pow((1 - t), n);
    Point2D tmp(0, 0);
    for (int i = 0; i <= n; i++) {
        tmp += points[i] * B;
        B *= (t / (1.0f - t)) * (1.0f / (i + 1.0f)) * (n - i);
    }
    //return Point2D(0, 0);

    return tmp;
}
```

实现效果


CGAssignment5: Bezier Curve


CGAssignment5: Bezier Curve

二、提升任务

实现更为简单、直观的de Casteljau算法来生成给定t值（t∈[0,1] ）下的Bézier曲线上的点。

实验原理

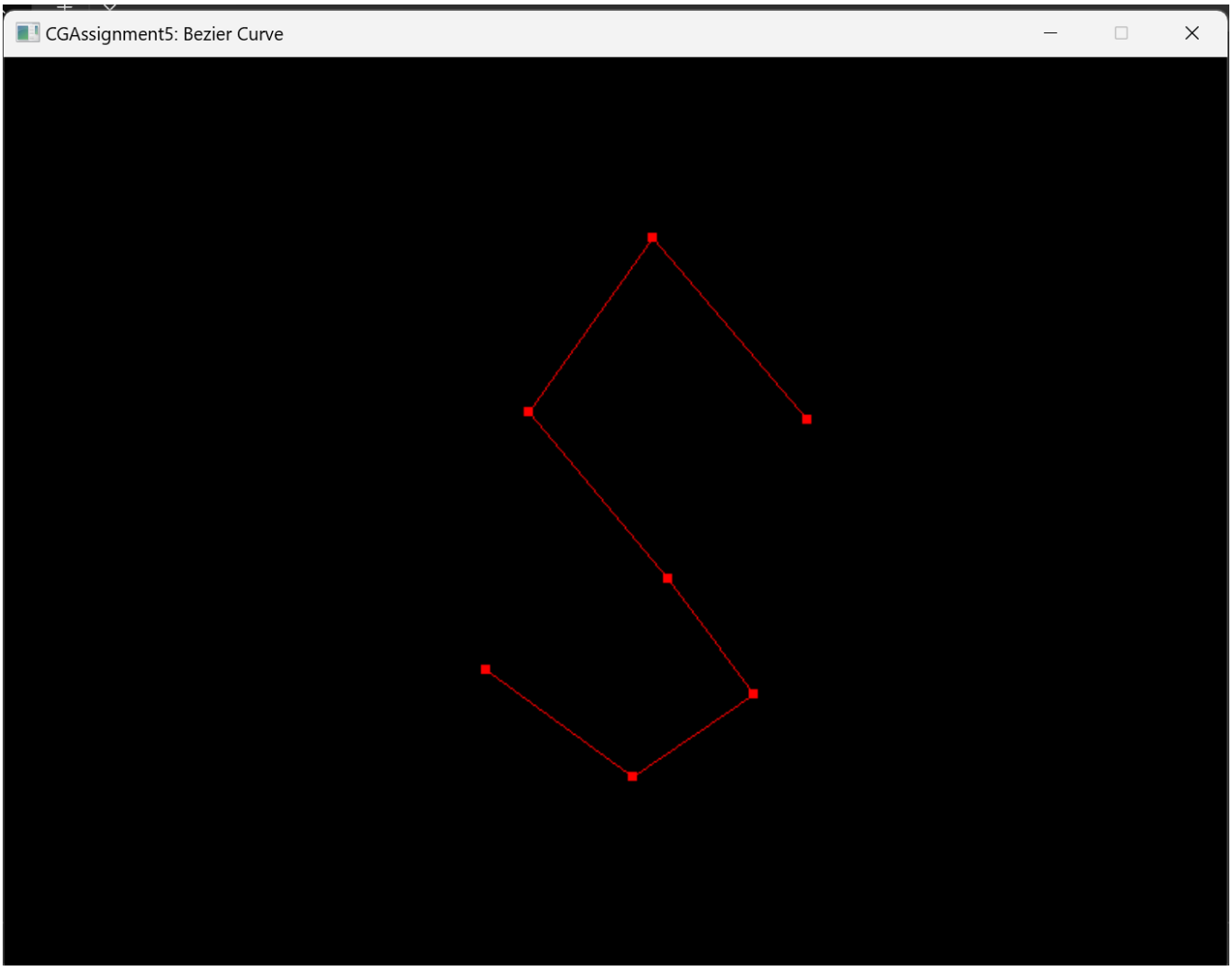
de Casteljau算法的核心伪代码可以描述如下：

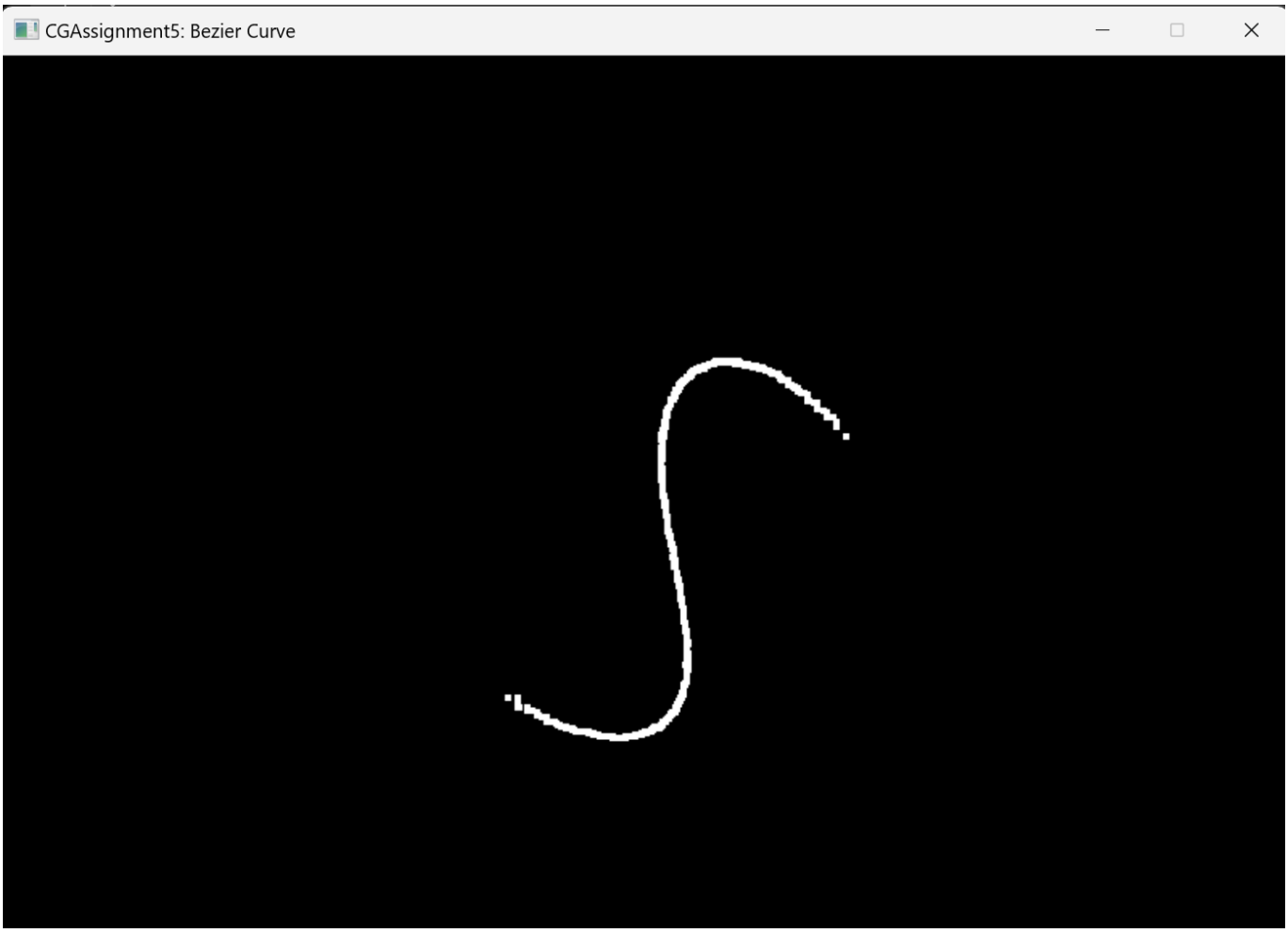
- 1、考虑一个$p_0, p_1, \ldots, p_n$为控制点序列的Bézier曲线，首先将相邻的点连接起来形成线段；
- 2、用$t:(1-t)$的比例划分每个线段，用线性插值法找到分割点；
- 3、对所有的线段执行上述操作，得到的分割点作为新的控制点序列，新序列的数量会减少一个；
- 4、如果新的序列只包含一个点，则返回该点，终止迭代过程。否则，使用新的控制点序列并转到步骤1，如此迭代下去。

代码实现

```
Point2D BezierCurve::improvementTask(const std::vector<Point2D> &points, const
double &t) const
{
    //improvementTask: implement de Casteljau algorithm for Bezier curve
    // Note: you should use Point2D::lerp().
    std::vector<Point2D> newControlPoints = points;
    while (newControlPoints.size() > 1)
    {
        std::vector<Point2D> tempControlPoints;
        for (size_t i = 0; i < newControlPoints.size() - 1; ++i)
        {
            Point2D splitPoint = Point2D::lerp(newControlPoints[i],
newControlPoints[i + 1], t);
            tempControlPoints.push_back(splitPoint);
        }
        newControlPoints = tempControlPoints;
    }

    return newControlPoints[0];

    //return Point2D(0, 0);

}
```

实现效果


CGAssignment5: Bezier Curve


CGAssignment5: Bezier Curve

三、挑战任务

在基础任务的基础上，实现对 Bézier 曲线的反走样。(对于一个曲线上的点，不只把它对应于一个像素，你需要根据到像素中心的距离来考虑与它相邻的像素的颜色。)