

課題の目標

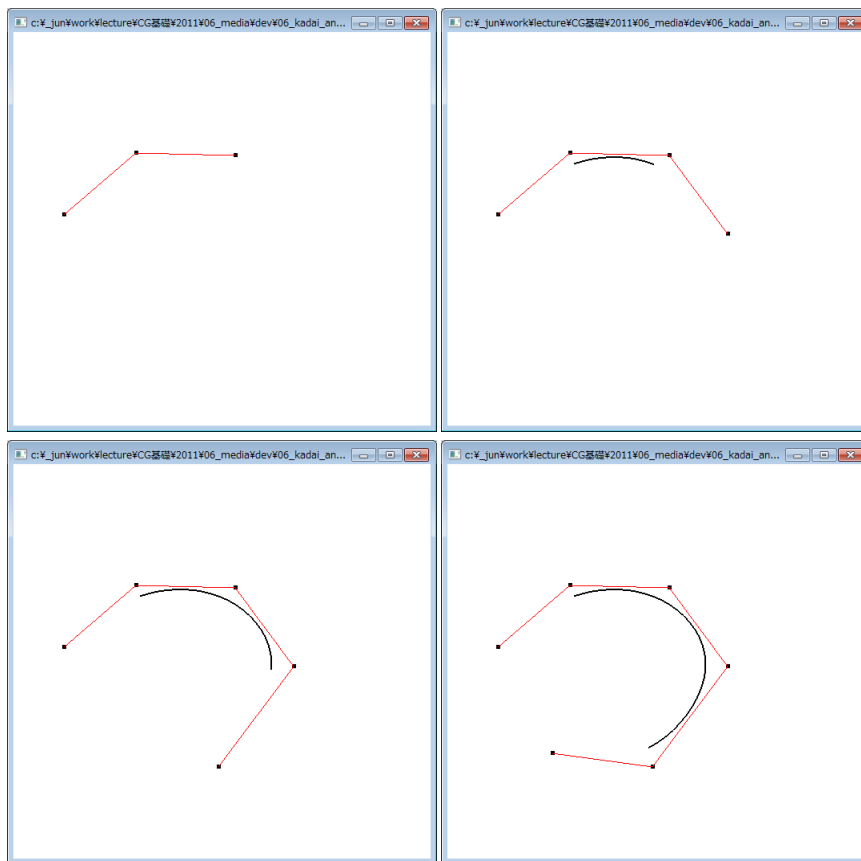
- ・パラメトリック曲線の **1** つである **B** スプライン曲線の仕組みを理解する
- ・制御点を入力することで、**B** スプライン曲線を描画するアプリケーションの開発を行う

課題の内容

- (1) サンプルコードに対して、下記の条件を満たすように **display()**関数を完成させなさい。プログラムができれば、なるべく滑らかに ∞ 記号を描いてみなさい。また、なるべく綺麗な円を描いてみなさい。

(条件)

1. マウスの左クリックで制御点を追加、右クリックで削減する (サンプルコードで実現済み)。
2. 入力された制御点を使って、**3** 次の **B** スプライン曲線を描画する。下図のように制御点の数に応じて複数のセグメントが連結された **B** スプライン曲線が得られるようにする (**3** 次なので **4** つの点が入力されるまでは何も表示されない。以降は制御点を追加するごとに **1** つのセグメントが追加される)。



- (2) プログラムができれば、ノットの数と値を次のように変更し、結果がどのようなになるか実験しなさい。

```
const int NUM_NOT = 8;  
double g_NotVector[] = {0, 0, 0, 0, 1, 1, 1, 1};
```

- (3) 上記以外にもノットの数と値を自由に変更し、それがどのように曲線に影響を与えるか実験し、考察を述べなさい。

(発展課題：オプション)

可能なだけ、以下の機能を追加してみる。

- ・一般化された N 次の B スプラインを描画するようにする。
- ・近似で構わないので、法線ベクトルを表示する。
- ・セグメントの境界がわかるように、セグメントの境界に点を表示する。
- ・基底関数のグラフを描画する。
- ・後からマウスドラッグで制御点を移動できるようにする (高難易度)
- ・ノットベクトルの位置を後から変更できるようにする (高難易度)