

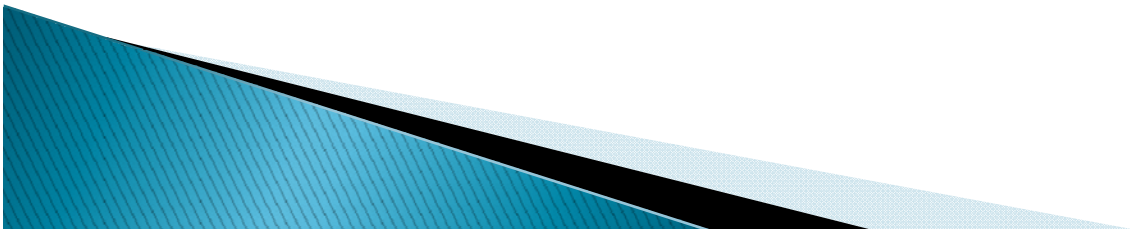
# CS3241 : Let's Get Twisted

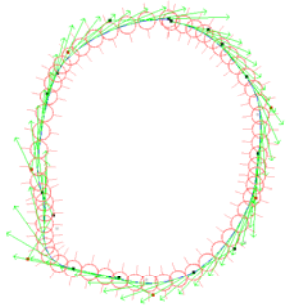
Lab #4



# Goals

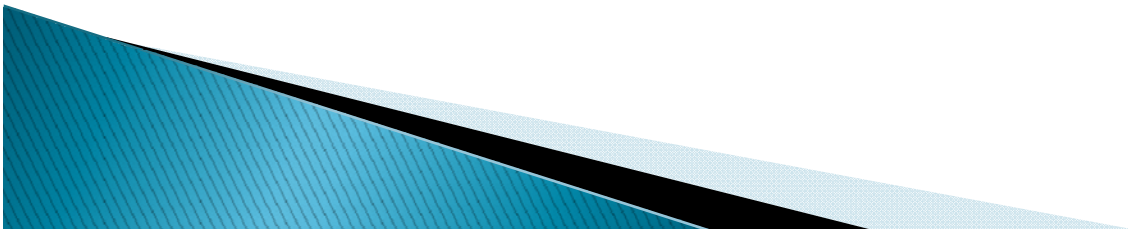
- ▶ Understanding Bezier Curves
  - Drawing by iterative method
  - Understanding tangent vectors and C1 continuity





# What you are given

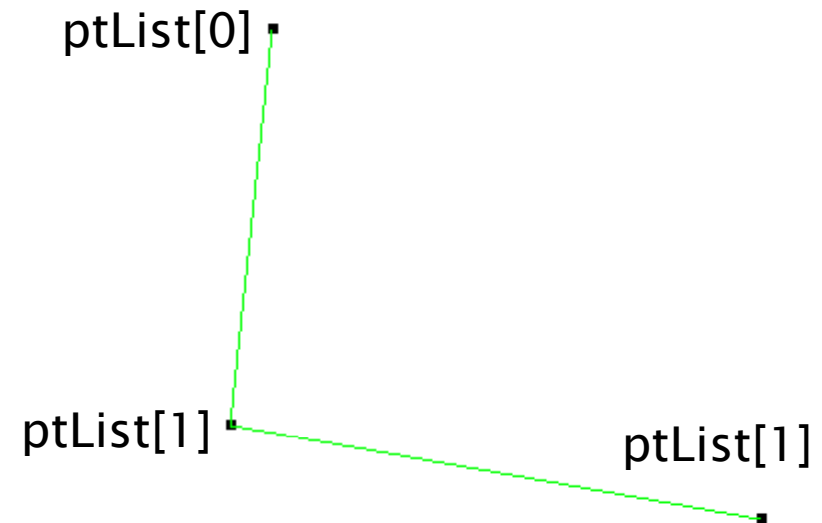
- ▶ A skeleton `main.cpp` file:
  - GLUT mouse handler to add points on screen
  - Reshape function to resize the screen
  - File handlers to read and write input/output files to save or retrieve the points

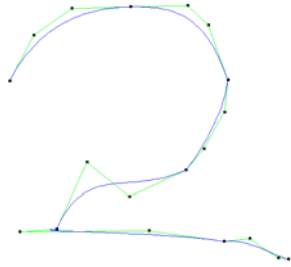




# Draw Control Lines

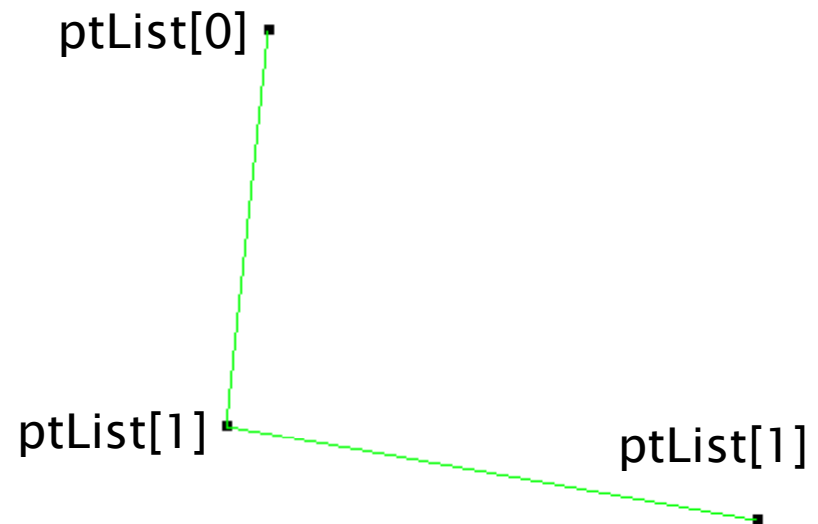
- ▶ Connect the adjacent points
- ▶ Placeholder provided in the `display()` function

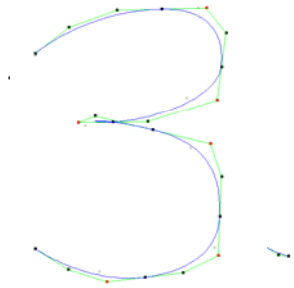




# Implement Erase Function

- ▶ Remember to implement 'Erase' function
- ▶ What is the easiest way to “eliminate” all the points ON THE SCREEN?



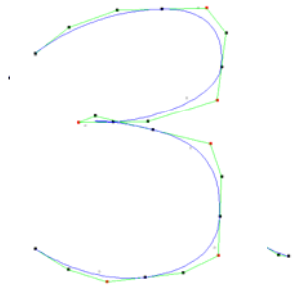


# Draw Curve

- ▶ Connect the 4 points in a curve by drawing NLINESEGMENT smaller lines
- ▶ Use the iterative method shown in lecture
- ▶ Order of the curve:  $n=4$
- ▶  $t$  ranges from 0 to 1 in NLINESEGMENTS (default of 32) steps

$$p(t) = \sum_{i=0}^n (b_{n,i}(t)) p_{0i}$$

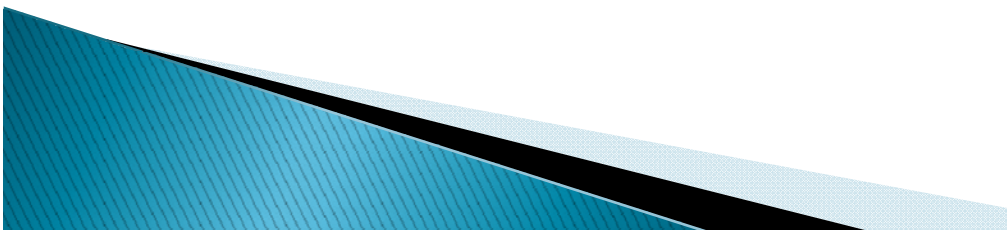
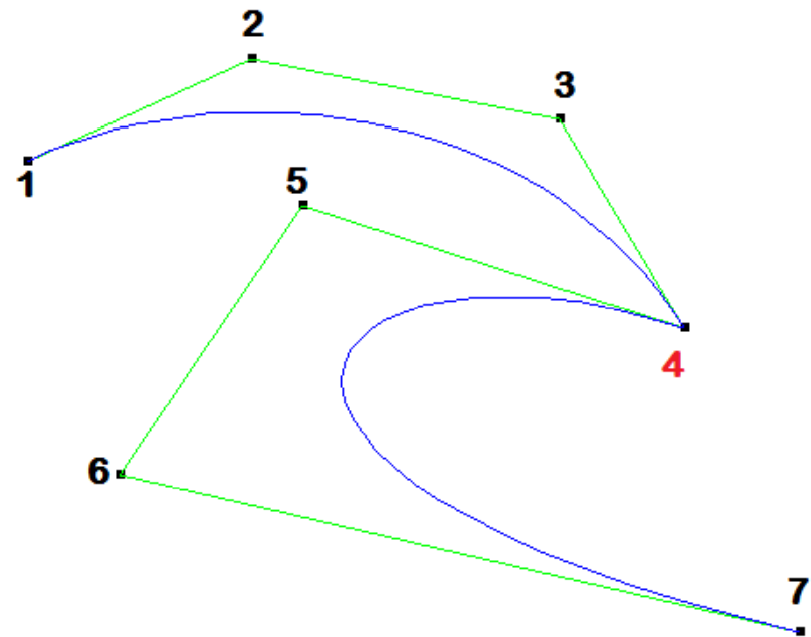
$$b_{n,i}(t) = \binom{n}{i} (1-t)^{n-i} t^i$$

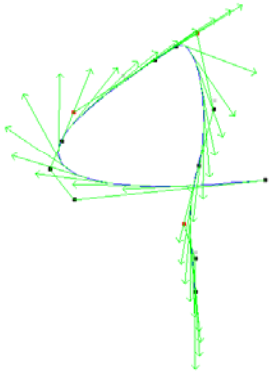


# Draw Curve

Reminder:

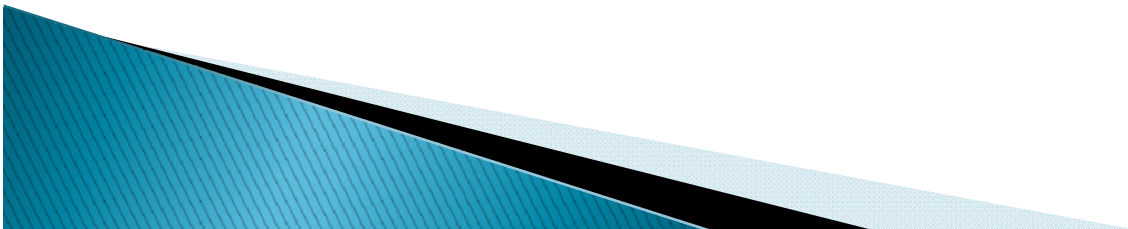
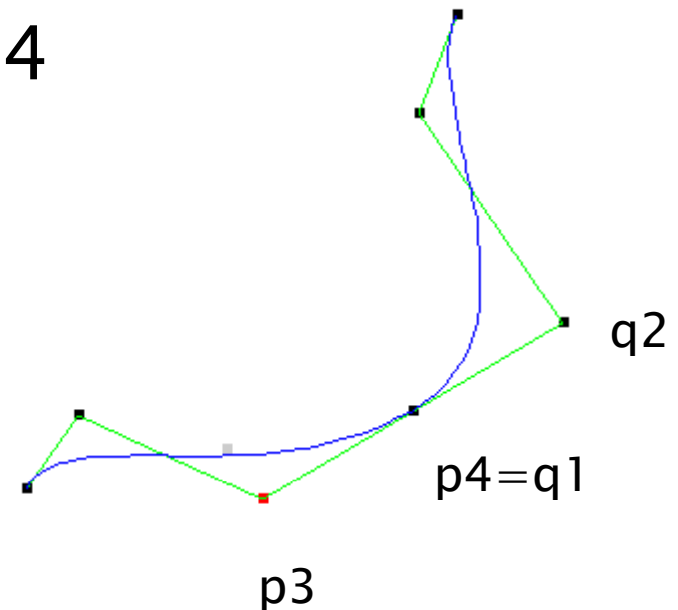
- ▶ Use Points 1–4 for first point,
- ▶ Points 4–7 for second etc as each adjacent cubic Bezier curve section has a common start/end point.



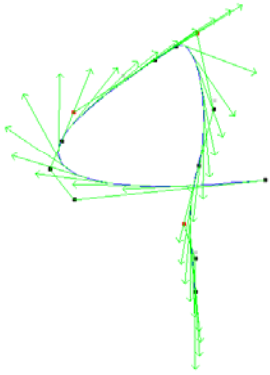


# Implement C1 Continuity

- ▶ For two curves (with same degrees) with control points  $\{p_1, p_2, p_3, p_4\}$  and  $\{q_1, q_2, q_3, q_4\}$
- ▶ They are C1 Continuous if
  - ▶  $p_4 - p_3 = q_2 - q_1$
- ▶ We know  $p_3$  and  $p_4$
- ▶ How to find  $q_2$ ?



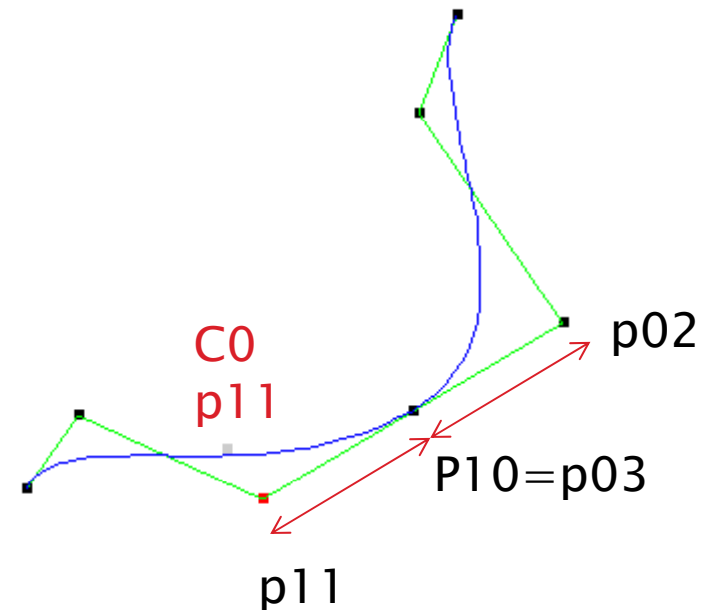


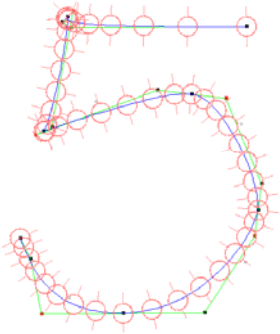


# Implement C1 Continuity

Remember:

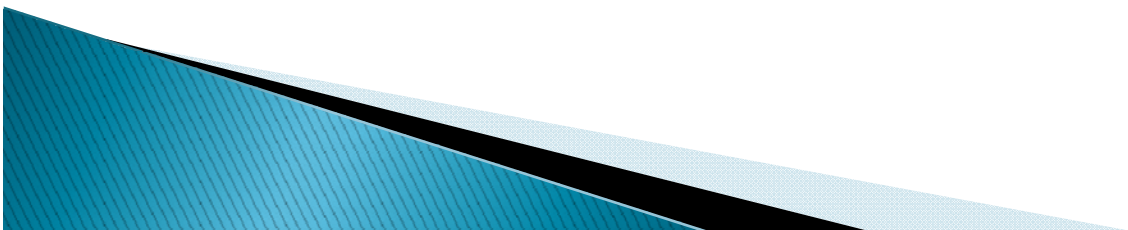
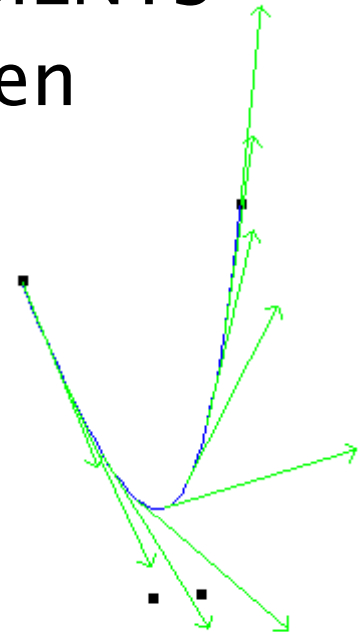
1. Draw the grayed out C0 point
2. The C0/C1 transformation must be reversible
3. All functions must work in either mode.

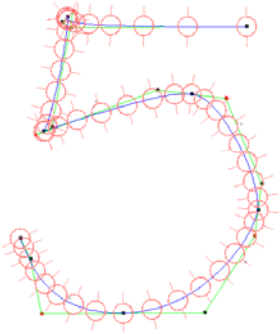




# Draw Tangents

- ▶ Use differentiation of Bezier function to compute tangent at each step
  - ▶ NOOBJECTONCURVE (default 8) objects – NOT NLINESEGMENTS
- ▶ `drawRightArrow( )` has been provided





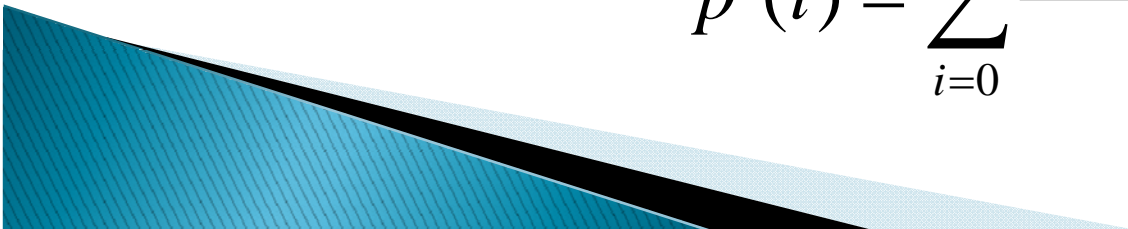
# Tangent revised

The tangent represents the velocity at the particular point.

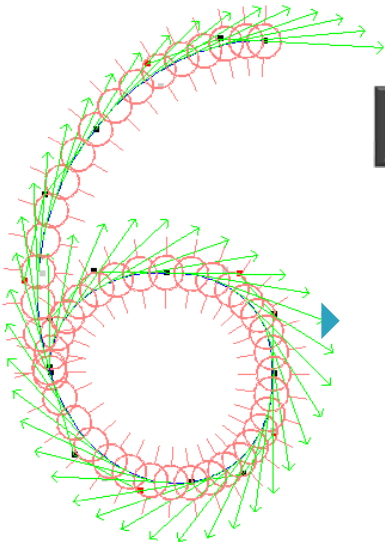
- ▶ The tangent is obtained by differentiating the equation for the point

$$\frac{d(p(t))}{dt} = \frac{d \sum_{i=0}^n b_{n,i}(t) p_{0i}}{dt}$$

$$p'(t) = \sum_{i=0}^n \frac{d(b_{n,i}(t))}{dt} p_{0i}$$



# Draw Object

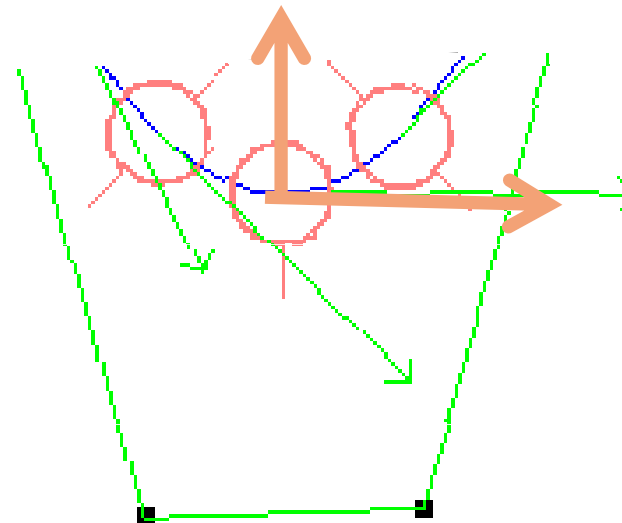


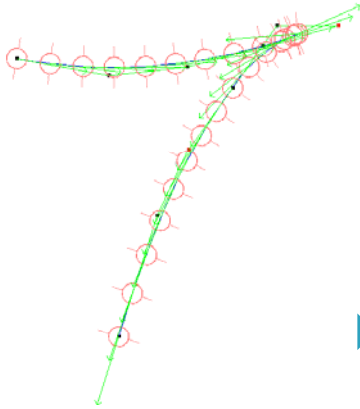
Draw any object you like along the curve

- ▶ NOOBJECTONCURVE (default 8) objects – same as tangents
- ▶ Objects must be oriented relative to tangent



You can reuse  
your Lab1 result!





# Final Drawing

- ▶ Draw a beautiful figure with a Bezier curve and your object.
- ▶ Default name is “savefile.txt”

