Problem Description

In this lab, you will write a simple curve modeling tool. Your tool must provide an interface for specifying a control polygon and then displaying a curve modeled by that control polygon by selecting one of three different techniques:

1. Bézier Curves: curve drawn using sampling and repeated linear interpolation with the de Casteljau method.
2. Bézier Curves: curve drawn using subdivision of the control polygon.
3. One other curve type of your choice, see below.

At a minimum, the user must be allowed to specify the control polygon, generate the curve using one of the above techniques, and then be given the option to edit the control polygon and regenerate a new curve. For the first method, the user should be allowed to specify the number of samples taken, whereas for the second method the user should be allowed to specify the number of subdivisions. You may need to include other options, depending on which of the following curve types you select for the third:

- Cubic B-spline curves with uniform knot vectors.
- NURBS
- Arbitrary order B-splines with arbitrary knot vectors
- Catmull-Rom Splines
- Cubic Bézier enforcing $C^1$ continuity at junctions (by specifying only a subset of the control points and repositioning the remaining)

Extra credit will be given for any extensions and/or new interfaces, including but not limited to:

- Providing the ability to specify duplicity of vertices in the control polygon (e.g. double, triple, etc.). An interesting interface might be involved here.
- Allow editing the control polygon while updating the curve simultaneously, specifically optimizing the technique so that the entire curve does not have to be recomputed.
- Implementing an additional types of curve formulations from the above list or otherwise.
- Tools to close the curve.
- Tools to elevate the degree.
- Tools for specifying tangent constraints in addition to positional ones.
Submission

(Please read all of these instructions carefully.)

There are no requirements for the language or platform that you choose to use to complete this assignment. Consequently, submission will require two components. First, I will reserve a block of time for you to demonstrate the assignment to me on the Monday after the due date. You will have 15 minutes to show me the features of your code and explain how they work.

In addition, by midnight the night before, you will submit your code on handin, in as compact a form as you can. I recommend creating a .zip file or a .tgz that packages up all of the necessary pieces. Be sure to include a README that explains both how to build and how to use your code. If you think I may need something special to build your code, feel free to ask in an email.

Submit using the handin procedure outline at https://handin.cs.clemson.edu/. You are welcome to use the commandline interface, but the web interface is sufficient. The assignment number is lab1.