Project 1: Keyframe Interpolation

Due: Jan 26, 2018, 11:55PM

In this project, you will implement an interactive program to demonstrate three different types of interpolation. The program lets the user add 2D control points on the screen by left-clicking the mouse and delete the most recently added control point by right-clicking. The resulting curve will be updated immediately on the screen after any UI event. You will implement the following three curve/spline types:

Cubic Bézier (splined together with C0 continuity, you will need 4 + 3 * i keyframes, where i is an arbitrary nonnegative integer)
Catmull-Rom
B-spline (defined by de Boor points)

The user can also switch between types of curves by using keyboard:
- 'l': linear interpolation (provided by the skeleton code)
- 'b': Bezier curve
- 'c': Catmull-Rom
- 'd': B-spline using de Boor points
- 'C': Remove all control points

Skeleton code
The skeleton code handles basic UI and visualization for you. After you successfully build it, try it out by giving different UI commands. You will find that the only interpolation that is working is the linear interpolation. However, the functionalities of left-click, right-click and keyboard control are already implemented for you.

Installation on Linux and Mac:
1. Download the skeleton code for mac and linux.
2. In terminal, go to the directory containing the downloaded code.
   cd 'path_to_downloaded_skeleton_code'
3. Unzip the code and go inside the directory.
   unzip Project1_Mac-linux.zip
   cd cs4496Proj1
4. Compile and run the code.
   make
   ./main.out

For Linux users: If the compiler says it can't find GL/gl.h, try installing freeglut library by:
Ubuntu# sudo apt-get install freeglut3-dev
or
Fedora/RedHat# sudo yum install freeglut-devel

If you have any questions or issues, contact the TA:

Installation on Windows:
To work on this and all subsequent projects for this class, it is advised that Windows users use Visual Studio, specifically VS2015. You can get a free version of Visual Studio Community 2015 by signing up on the Microsoft Dreamspark website (It also is the default editor for Unity 5.3.* now, replacing Mono, from what I have seen, so you may have it already if you have a current install of Unity) :

https://www.dreamspark.com/Student/Default.aspx

This download is a fairly large (3.5 gb) disk image (*.iso) which you will have to mount – right-click the .iso file and select “Mount” and then select “OK” on the dialog. This will mount the .iso (treats it like a disk) and show the file structure on the disk image. From here just select vs_community.exe and follow the directions. Be sure to select Custom install, and then under languages select “C++”. This may take a while, it updates the installation as it installs.

Remember to unmount the disk image when you are done – right-click the .iso and select “Unmount”.

Once you have downloaded and installed VS2015, you should be able to open the cs4496P1_sln.sln file that is part of the assignment template code. This will open the development environment. You can switch between “debug” mode (supports break points) and “release” mode (runs faster) by selecting from the drop down in the toolbar. If you have any questions or issues, contact the TA:

Extra points
1.For each curve, you can implement support for "wrapping," which means that the curve has C0 continuity between the end point and the beginning point both in x and y coordinates (1 point).
2.Implement a C2-Interpolating curve (3 points).
3.Support UI that modifies the position of the control points by click-and-drag (1 point).
4.Support UI that deletes any control point by right-clicking on it (instead of always deleteing the last point) (1 point).
5. Make b-spline start and end at the first and last de Boor points (1 point).

Submission Instructions
Mac Users:
1) Clean up the folder using "make clean"

2) Zip the directory containing your code and submit this zip file to T-Square

Windows users:
1) Zip the folder containing the .sln file.

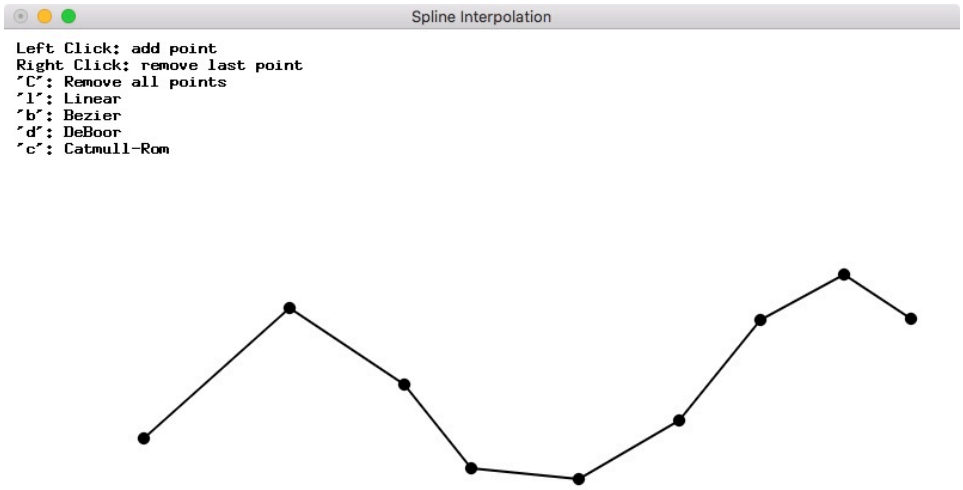
2) REMOVE the following files and directories from the .zip file :
- cs4496P1_sln.sdf
- cs4496Win_Project1\GL
- cs4496Win_Project1\Eigen
- cs4496Win_Project1\Release
- cs4496Win_Project1\Debug

Your zip should be no larger than 5 MB in size. If it is bigger than this, you need to make sure you have removed all the aforementioned files and directories from your zip file.

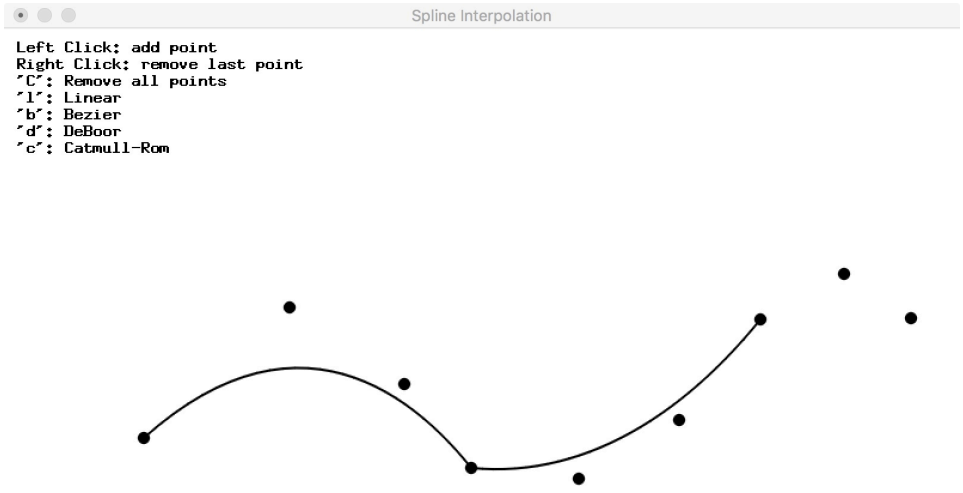
3) Submit this .zip file to T-Square.

Solution examples
Here are a few images to demonstrate what each correct curve should look like for one example set of control points. You can qualitatively compare them with your solution.

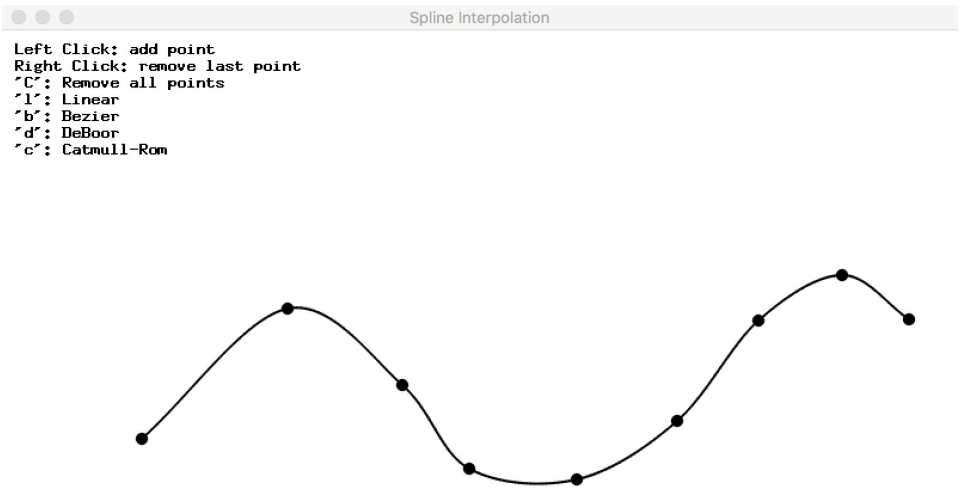
Linear:



Bezier:



Catmull-Rom:



B-spline: