

# CAS 737 Computer Animation Winter 2024

## Assignment 1

### Curves - Catmull Rom Spline

In the class, we derived **cubic** Bezier basis and showed how to compute point positions on the curve. Here an example python code implementation of **cubic** Bezier curve is provided, for you to have an easy start.

### Tasks

The tasks for assignment 1 are:

1. Run the provided python code, get familiar with the UI control and make sure the Bezier curve works properly.

Left mouse click - add control points

Right mouse click - clear control points

Keyboard button 'p' press - capture screenshot in ppm image format

2. Add your own implementation for Catmull-Rom spline.

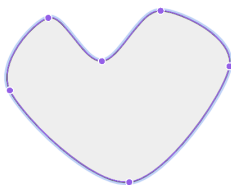
2.1 Derive Catmull-Rom basis. Make sure the basis is correct before coding. **30'**

2.2 Use the basis to implement Catmull-Rom curve in the provided python file. **50'**

No need to group 4 control points (as for **cubic** Bezier) to draw one curve segment.

Two neighboring control points should be able to define a cubic Catmull Rom segment. Figure out how to handle the derivatives for the very first and last control points.

3. Draw something interesting using your implemented Catmull-Rom, capture at least two different ppm images. **10'**
4. Support closed-loop curve as exemplified below in code. **10'**



# Submission

Submit the following files in a zip through Avenue:

- A pdf file with the derived Catmull Rom Basis in it
- Source python code with your implemented Catmull Rom Curve
- Two designed images drawn with your own implemented Catmull-Rom in ppm format

Deadline: Feb 10, 2024