

CPSC 426: Computer Animation

Assignment 1 (10%)

out: Fri January 17, 2014
due: in class Wed January 29, 2014

Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.

Name: _____

Student Number: _____

Question 1	/ 5
Question 2	/ 5
Question 3	/ 5
Question 4	/ 2
Question 5	/ 7
TOTAL	/ 24

This assignment has 5 questions, for a total of 24 points.

1. (5 points) Math review

$$a = \begin{bmatrix} -2 \\ 3 \\ 1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ -1 \\ 5 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 3 & 3 \\ 5 & 1 & 1 \end{bmatrix}, d = 2$$

For each of the following, compute the answer or, if it cannot be evaluated, state that it is a nonsense expression.

$$a^T b$$

$$a b$$

$$a b^T$$

$$b^T a$$

$$a d$$

$$d C$$

$$b C$$

$$C a$$

$$a^T C a$$

$$b^2$$

2. (5 points) Use the `benettonplay` website to create a flipbook animation of between 30 and 100 frames in length about life in Vancouver or at UBC. This could be an event, an activity, something inspiring, something frustrating, etc. Animations will be scored based on their overall appeal, story, image quality, and motion quality. This will necessarily be subjective. Remember to have fun with it! This is not an art course and so the expectations will be set appropriately.

`http://www.benettonplay.com/toys/flipbookdeluxe/guest.php`

Create the animation either as a guest or a registered user. As a guest user, your animation can be no longer than 100 frames. A simple user tip: make good use of the “insert copy of current frame” button that is found to the right of the “right arrow”, which can also be used to create a new (blank) frame. Some users have reported problems with saving longer animations, so experiment with short and medium length animations first. We will show a selection of the final animations in class. If you like, you can also export your animation frames as PDF, which will print them in a way that lets you cut them out to build your own flipbook.

When you “save and publish” your animation, **be sure to include cs426** as part of your name and as part of the title you give your animation. To see the final URL that is assigned to your flipbook, select the “Embed in your site” option.

Give the following information for your animation:

- (a) URL for your flipbook animation: _____
- (b) Title of your flipbook animation: _____

3. (5 points) Develop a cubic parametric curve where the user specifies the starting ($t=0$) position and velocity, the midpoint ($t=0.5$) position, and the endpoint ($t=1$) position. Build your geometry vector using this given order for those elements. Use Matlab or one of the many online matrix calculators to perform the matrix inversion (Google “online matrix inversion”). As your solution, give the “final form” for the $x(t)$ curves, as done for the examples in class. Also give the basis functions and give a sketch or plot of what they look like.

4. (2 points) A user creates a curve using Bezier control points, G_{bez} .
- (a) How would one compute the B-spline control points, $G_{bspline}$, that represent the same curve?

(b) **optional bonus (2 points):** Suppose that we wished to subdivide the Bezier curve, $x(t)$, into two Bezier curves, $x_1(u)$ and $x_2(v)$, such that $x_1(u), u \in [0, 1]$ exactly traces out the curve $x(t), t \in [0, 0.5]$, and $x_2(v), v \in [0, 1]$ exactly traces out the curve $x(t), t \in [0.5, 1]$. Show how one would compute the Bezier control points for the subdivided curves, i.e., G_1 and G_2 , given the original G_{bez} control points.

5. (7 points) Implement a basic OpenGL program that demonstrates the use of Catmull-Rom splines to control the 3D movement of a ball through a simple scene. The control points should be read in from a text file. The demonstration should repeat the playback in a loop. Use regular sampling of the parametric curve parameters $t \in [0, 1]$ in order to display the frames of the animation. Thus taking a smaller time step in t would produce more frames for the same motion, and therefore a slower apparent motion of the ball. Your motion should consist of at least 10 spline curve segments, C_i that each cover a successive $1s$ interval.

You can begin from any OpenGL starter code that you like. Basic starter code will also be provided for the lab Linux machines in ICCS 005. It is also fine to develop code on your own machine or laptop. The only caveat is that we can then provide more limited support in terms of helping with issues that are specific to your machine. In this case, you should still submit it using the lab machines, along with a note in the README.txt file that notes whether or not it runs on the lab machines. We will aim to do one-on-one marking for the implementation component of assignments.

optional bonus (up to 5 points): Build a simple articulated hopping lamp and animate it's motion using either piecewise Hermite curves or Catmull-Rom curves to define the motion for all its degrees of freedom. Using simple text files to define the motions is fine. The motions will be easiest to design by using keyframes to specify the position of all the degrees of freedom at given times. You may want to implement the ability to place keyframes at arbitrary points in time.

Do not hand in any printed code. Create a `README.txt` file that includes your name, student number, and login ID, and any information that you would like to pass on the marker. Create a folder called 'a1' under your cs426 directory and put all the source files, your makefile, and your `README.txt` file there. Do not use further sub-directories. The assignment should be handed in using:

```
handin cs426 a1
```