# MP1: Victory Dance

**Assigned: August 31, 2016.**      **Due: 11:55pm Sept. 16, 2016.**

For your first Machine Problem, you will create a 2-D animation of a dancing Illinois Victory Badge (as shown above). To complete the MP, you will need to create a digital model of the badge, write code to render it, and write code to animate it.

## Modeling

You will need to model the badge with a 2-D mesh of triangles. You can use more than one mesh; you could create a mesh for the blue part and then separate meshes for the orange bars. Or you could do something different...it just has to look like the badge. The mesh(es) must not contain any "T" intersections, which means that there must be a vertex located at any junction in the mesh. You can also model a background, or just use a solid background color. One approach to creating the mesh would be to get some graph paper, draw the badge and figure out a set of coordinates for the vertices and edges for the triangles that works. Once you have the coordinates and triangles, just write up a set of JavaScript arrays in your code containing those numbers.

## Rendering

You should color the badge and the background as shown above. For this MP, you can render using the gl.drawArrays call, with the primitives specified as gl.TRIANGLES. Note that this will draw a triangle for each group of three consecutive vertices. For example, 12 vertices create 4 separate triangles. This means that the coordinates of vertices shared by multiple triangles will be repeated in the buffer. If you wish, you can use a different implementation (it just has to work), but the one suggested here is probably the simplest.

## Animation

You will need to write a program to change the location of vertices over time to animate your model. Your code should use two different methods for changing the vertices:

1. Use affine transformations (such as scaling, rotation, or translation). These should be implemented as matrix transformations (you can use the glMatrix library).
2. Use a motion that cannot be implemented as an affine transformation (e.g. have the orange rectangles wiggle like snakes). This part of the animation could be data driven using a table of pre-defined vertex positions for the motion. The motion can also be keyframed, so the vertices are interpolated from one keyframe location to a second keyframe location. Linear interpolation is fine for this assignment, but other interpolations, such as an s-curve, will ease the figure in and out of each keyframe. As an example of the kind of motion you might consider, imagine using various

combination of sine waves to generate the dancing motion. If you modify the vertex positions by changing the coordinates in the buffer, make sure you use gl.DYNAMIC_DRAW.

# Submission

You will need to submit the following file in a zip archive via Compass:

mp1.html
mp1.js

All of your original code should be in those 2 files. You should also include any additional files you use. In particular you can use gl-matrix and the WebGL utilities used in the examples in class:

- Utility code: webgl-utils.js
- glMatrix library: gl-matrix-min.js

# Grading

The assignment will be graded based on the following scale:

50% Program runs without crashing, sucessfully creating a WebGL context and rendering multiple frames.

10% Code is well documented, with comments for each function explaning what that function does.

10% Badge model looks substantially like the provided image.

10% Badge model is a set of triangle meshes and the meshes have no t-junctions.

10% Affine transformations used in animation.

10% Animation includes some component that does not use affine transformations

**Name your submission as ${NetID}_MP1.zip and submit it to Compass.** This MP (as are the other MP's) is more than just an assignment for a grade to verify graphics programming proficiency. It is also an opportunity to impress me, your friends/family and your future employer or graduate school. These assignments are designed to provide the student with a portfolio of graphics programs and associated images and animations that can be used to demonstrate graphics expertise. Many students ask me to write letters of recommendation for jobs or grad school and I base these recommendations on their MP results. Use this MP as a chance to show off your graphics skills and make a strong impression.