Project A: Waving Arm and Rotating Boxes

Evan Allan (eah8003)

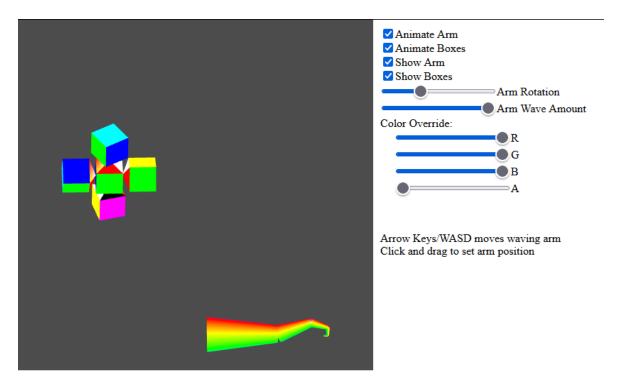
Goals

- Create interesting movements using simple shapes
- Create movements that look fluid, as if it were a single object bending rather than multiple objects rotating around points in space
- Use blending to demonstrate the ability to mix colors in the shader program

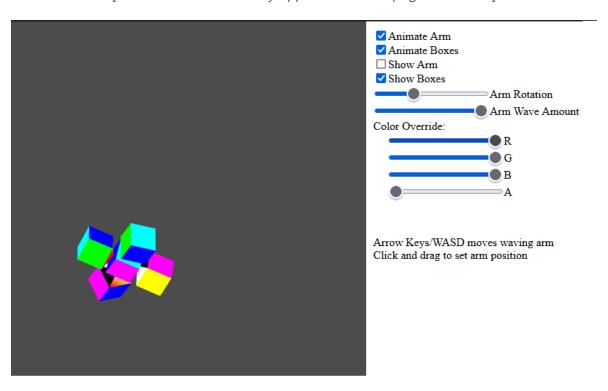
Controls

- Pressing arrow keys or WASD moves the waving arm around the screen.
- Clicking sets the position of the waving arm.
- Clicking and dragging moves the waving arm around the screen.
- "Animate" checkboxes enable/disable waving/rotating animations.
- "Show" checkboxes show/hide objects on screen (note that animations will still happen in the background so when an object reappears it may not be where it was when it was hidden).
- "Arm Rotation" slider rotates the arm around the X axis so you can see all the parts of it.
- "Arm Wave Amount" slider changes how wavy the arm is.
- "Color Override" sliders allow the user to override all vertex colors with a custom RGBA value. The selected alpha value will be used for blending with the original color.

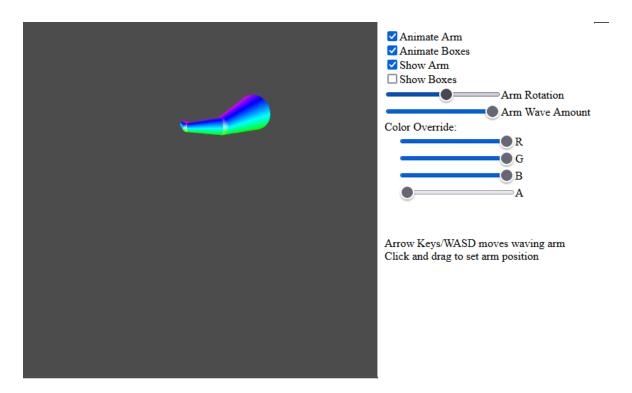
Results



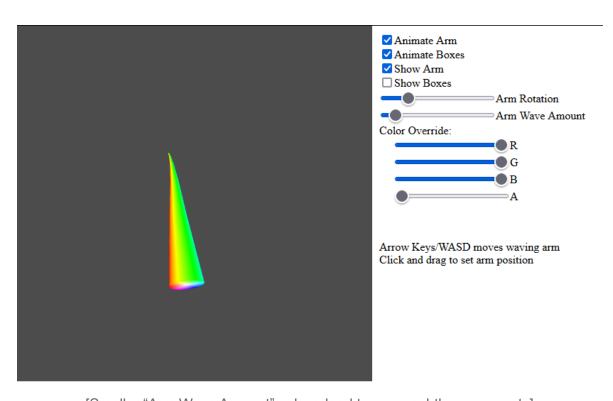
[Both assemblies as they appear when the page first loads]



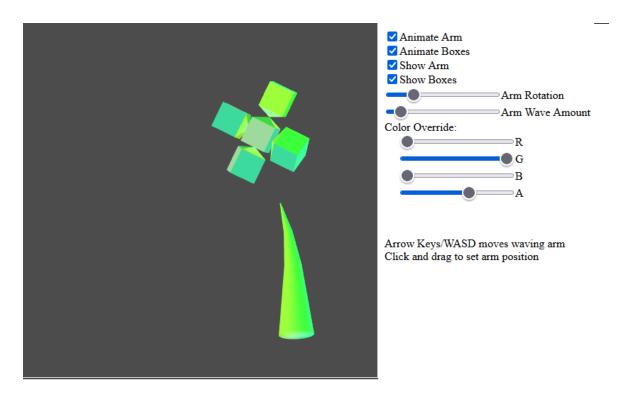
[The boxes rotate around many axes and move around the screen]



[The arm can be moved and rotated about the x axis]

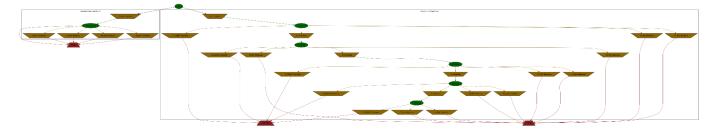


[Smaller "Arm Wave Amount" values lead to more subtle movements]



[Color Override can be used to apply a tint to all vertices]

Scene Graph



The scene graph is a bit large, as seen above. A bigger version exists at https://bit.ly/3bcTDEu. The graph creator I was using does not support the T shape for transforms, so I replaced it with a trapezoid with the shorter side on the bottom, which almost resembles the T shape. The squished cylinder object (for each of the sections of the waving arm) is split into multiple meshes - the circles at the top and bottom, and the curved rectangle around the middle (which is named "Cylinder" in the graph). This makes generating points easier. A single function generates transforms for the Top, Bottom, and Middle portions, so they will never be misaligned, and so they can be treated as a single part.

This graph was created based on the object hierarchy, which is shown below:

```
Name Graph: ▼ Object { root: {_}} }
           ▼ root: Object { 11: {_}}, house: {_} }
              ▼ house: Object { house2: {}, house3: {}, house4: {}, _ }
                 ▶ house2: Object { }
                 ▶ house3: Object { }
                 ▶ house4: Object { }
                 ▶ house5: Object { }
                ▼ 11: Object { 11_Top: {}, 11_Bot: {}, 11_Middle: {}, _ }
                ▶ 11_Bot: Object { }
                 ▶ l1_Middle: Object { }
                 ▶ 11_Top: Object { }
                 ▼ 12: Object { 12_Top: {}, 12_Bot: {}, 12_Middle: {}, _ }
                    ▶ 12_Bot: Object { }
                     ▶ 12_Middle: Object { }
                     ▶ 12_Top: Object { }
                     ▼ 13: Object { 13_Top: {}, 13_Bot: {}, 13_Middle: {}, _ }
                        ▶ 13_Bot: Object { }
                         ▶ 13_Middle: Object { }
                         ▶ 13_Top: Object { }
                         ▼ 14: Object { 14_Top: {}, 14_Bot: {}, 14_Middle: {}, _ }
                            ▶ 14_Bot: Object { }
                            ▶ 14_Middle: Object { }
                            ▶ 14_Top: Object { }
                            ▼ 15: Object { 15_Top: {}, 15_Bot: {}, 15_Middle: {} }
                               ▶ 15_Bot: Object {
                               ▶ 15_Middle: Object { }
                               ▶ 15_Top: Object { }
                              ▶ <prototype>: Object { _ }
                           ▶ <prototype>: Object { ... }

                 ▶                                                                                                                                                                                                                                                                                                                                                    <pre
```