Student ***NetID***:_____ Name:_____ Grader Name:_____
(netID == 3 or 4 letters, 3 or 4 digits: e.g. JET861 Please write clearly; make it easy to read)

CompSci 351-1 Grading Sheet Project A Fall 2022

J. Tumblin 4/3/2022

_____**5% All file-naming correct, with clear illustrated PDF report** with name, netID, title, goals, help, user-guide, ≥4 results pictures, and an (optional) sketch of your program's scene-graph (transform tree).

_____ **5% Sensible, Complete On-Screen User Instructions:** From the program's on-screen display, even new users can quickly and easily identify and use all your program's features and options without your help.

_____**10% At least two different rigid 3D parts that YOU designed,** more complex than a rectangle or cube (>12 vertices), each made by drawing from contents of a Vertex Buffer Object (VBO).
(HINT: Make your own drawing fcns, *e.g.* drawHexa(), drawRobot(), drawBicycle(), …)

_____**10% Rasterized per-vertex colors-everywhere:** All vertices for all rigid 3D parts must be stored in the VBO, and all must include position attributes and RGB color attributes (see Chapter 5). In each rigid 3D part, one or more triangles must have 3 obviously-different vertex colors (not just 2!). No colors change with position of the part or the assembly (e.g. no lighting effects!).

_____**10% Traveling Assembly**: At least one entire assembly is not stationary – it 'travels' continuously on-screen without requiring any user interactions (e.g; a butterfly continually moves among a set of flowers)

_____**10% Flexing/Spinning Joints:** All joint-angles of at least one assembly must continually change, keeping its rigid 3D parts moving smoothly and continually. Joints may spin (e.g. always-growing rotation angle) or flex (rotation angle grows then shrinks cyclically) or move in other ways too.

_____**20% KINDS: Two or more obviously-different *kinds* of assemblies of rigid 3D parts.** Each *kind* has a different/dissimilar scene-graph shape (thus different joint sequences); each *kind* draws each of its rigid 3D parts using dissimilar matrix transforms (causing obviously different movements), and these movements are obviously not synchronized (e.g. visibly different cycle times for periodic movements).

_____**10% At least one kind of assembly must have two or more sequential, moving joints**, with the two sequential joints at different 3D locations. (e.g. robot head that turns, nods, and tilts is still just 1 joint; arm that rotates at shoulder and elbow is 2 joints at different 3D locations). Only 1 joint location? → half-credit.

_____ **10% Keyboard Interaction:**
One or more on-screen part(s) or assembly(ies) change visibly & obviously in response to various keyboard inputs.

_____ **10% Mouse-Drag Interaction:**
One or more on-screen part(s) or assembly(ies) make on-screen movements that match mouse-drag amounts. For example, translate an assembly by the amounts of mouse-drag (see ControlMulti starter code for rotate).

_____ **EXTRA CREDIT:**
_____up to 2%: add webpage controls & features not found in 'ControlMulti' (sliders, menus; try dat.gui?)
_____up to 2%: user-adjustable color for one or more moving 3D part.
_____up to 2%: user-adjustable flex-angle (start angle, stop angle) for at least one rigid 3D part in an assembly.
_____up to 2%: accurate 'Scene Graph' diagram that includes all rigid 3D parts and all assemblies in the scene.

==========**TOTAL POINTS/100** (30% of final grade)