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

CompSci 351-1 Grading Sheet Project A Fall 2019

J. Tumblin 10/04/2019

_____**10% All file-naming correct + clear illustrated PDF report** with name, netID, title, goals, help, user-guide, ≥4 results pictures, and sketch of your program's scene-graph (transform tree).

_____ **5% 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 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% Smoothly-varying per-vertex colors:** All 3D parts consist of triangles that vary their colors between their vertices, using at least 3 obviously-different vertex colors (not just 2!), and interpolated by 'varying' variables in shaders. Every vertex must use RGB color attributes in every vertex in the VBO (e.g. proper use of 'stride' and 'offset'; explained in Chapter 5 and demonstrated in starter code).

_____**10% Animation**: On-screen objects move continually (movement requires no user actions).

_____**10% Two or more clearly-different *kinds* of objects.** Each *kind* will draw each of its rigid 3D parts with different matrix transforms (thus they move differently), & a differently-shaped scene-graph (thus their joints connect differently). Each different *kind* of object should move at different rates, independently, continuously, and not synchronized (e.g. different cycle times for periodic movements)

_____**20% Two or more sequential, moving joints** within each of these 2 different *kinds* of objects (with every joint at a different on-screen location. (Only 1 joint location? → half-credit)

_____1**0% Always-Smooth On-screen Movements:** All animation and all user-controls cause SMOOTH on-screen changes (locations, poses, sizes etc.). No large sudden 'jumps'!

_____ **5% Keyboard Interaction:**
On-screen objects move and change in response to various keyboard inputs.

_____ **5% Mouse-Click Interaction:**
On-screen objects move and change due to mouse clicks at different locations.
(excludes webpage buttons and mouse-drag interactions: objects must respond to clicks alone)

_____ **5% Mouse-Drag Interaction:**
On-screen objects move and change in response to mouse dragging in the canvas.
(mouse-drag includes button down/up events, but these don't count as mouse-click interactions)

_____ **EXTRA CREDIT:**
_____up to 2%: add better-looking webpage controls & features (buttons, menus; try dat.gui?)
_____up to 2%: automatic object **color-change** : smoothly, dramatically &visibly over time,
_____up to 2%: object/part **shapes** change smoothly dramatically &visibly over time,
(e.g. upper-arm segment changes length and/or width; lower arm segment tapers/bulges..)
_____up to 2%: Report includes accurate 'Scene Graph' diagram for each kind of object

==========**TOTAL POINTS/100** (24% of final grade)

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

EECS 351-1 Grading Sheet: Project A Win 2019

J. Tumblin 1/28/2019

_____**10% All file-naming correct + clear illustrated PDF report** with name, netID, title, goals, help, user-guide, ≥4 results pictures, and sketch of your program's scene-graph (transform tree)

_____ **5% 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 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% Smoothly-varying per-vertex colors:** All 3D parts vary their colors between their vertices, using at least 3 obviously-different vertex colors (not just 2!), and interpolated by 'varying' variables in shaders. Every vertex must have RGB color attributes in the VBO (e.g. proper use of 'stride' and 'offset' as described in Chapter 5 and demonstrated in starter code).

_____**10% Animation**: On-screen objects move continually (movement requires no user actions).

_____**10% Two or more clearly-different *kinds* of objects.** Each *kind* will draw each of its rigid 3D parts with different matrix transforms (thus they move differently), & a differently-shaped scene-graph (thus their joints connect differently). Each different *kind* of object should move at different rates, independently and continuously.

_____**20% Two or more sequential, moving joints** within each of these 2 different *kinds* of objects (with every joint at a different on-screen location. (Only 1 joint location? →half-credit)

_____1**0% Always-Smooth On-screen Movements:** All animation and all user-controls cause SMOOTH on-screen changes (locations, poses, sizes etc.). No large sudden 'jumps'!

_____ **5% Keyboard Interaction:**
On-screen objects move and change in response to various keyboard inputs.

_____ **5% Mouse-Click Interaction:**
On-screen objects move and change due to mouse clicks at different locations.
(excludes webpage buttons and mouse-drag interactions: objects must respond to clicks alone)

_____ **5% Mouse-Drag Interaction:**
On-screen objects move and change in response to mouse dragging in the canvas.
(mouse-drag does require button down/up, but these don't count as mouse-click interactions)

_____ **EXTRA CREDIT:**
_____up to 2%: add better-looking webpage controls & features (buttons, menus; try dat.gui?)
_____up to 2%: automatic object **color-change** : smoothly, dramatically &visibly over time,
_____up to 2%: object/part **shapes** change smoothly dramatically &visibly over time,
(e.g. upper-arm segment changes length and/or width; lower arm segment tapers/bulges..)
_____up to 2%: Report includes accurate 'Scene Graph' diagram for each kind of object

==========**TOTAL POINTS/100** (24% of final grade)