OVERVIEW:

- This an iOS App developed using Swift and GLSL in Xcode.
- It draws a cuboid with the x,y and z coordinate axes. This cuboid can be rotated about any axes passing through its centre.

EXECUTION :

- Open the .xcodeproj file in Xcode.
- Run the project after setting the active scheme as iPhone6
- The can be directed to an actual handset by selecting the connected iPhone as the active scheme.

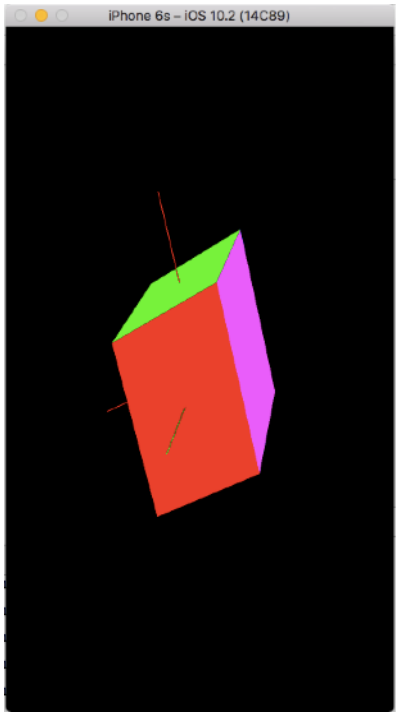
BEHAVIOR:

- When the app launches in simulator, there will be a cuboid on the screen with the three coordinate axes. Any touch moved on the screen will rotate the cuboid and the coordinate axes about an axis orthogonal to the direction of movement.
- The rotations are not persistent with subsequent touches.
- All the rotations are done at GPU using a modelling matrix which is multiplied with the position attributes in the vertex shader.

THE CODE

- In Xcode go to the GameViewController.swift from the Project Navigator. This file has the touch-listeners and basic functions like setupGL(), glkView(). The delta x and delta y calculated as a result of touch moved are passed as uniform variables to the vertex shader.
- The vertex shader forms a rotation matrix using these uniform variables in the rolling ball algorithm.
- The resultant rotation matrix forms the modelling matrix to calculate gl_Position.
- gl_Position is sent to the fragment shader.

SCREENSHOT:



LIMITATIONS:

- In this project no user inputs are taken to customise the modelling in terms of rotation, translation and scaling. It has only one transformation which is rotation.
- The model and view matrices are not separate which is not the correct implementation of graphics pipeline.

- There is only one pre-drawn object which can be modelled. It doe not have an option of importing new objects or having multiple objects on one screen.
- These limitations will be overcome in upcoming tasks.

: