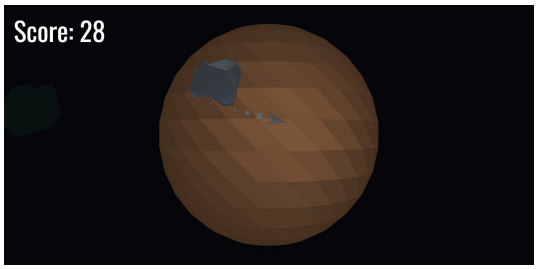
Project: Totally Not Asteroids (Dodger)
Group: Cole Chittim, Christopher Daw

Repo: https://github.com/ColeChittim/final-480-chittim_daw
Github Pages (Hosted Game): https://colechittim.github.io/final-480-chittim_daw/

Summary of accomplishments:

Our final deliverable expands upon assignment 3 by modifying the rendering engine to allow for user input via the arrow keys or on screen controls to move around a 3D player object and dodge other objects that appear and move on screen. If you collide with one of these objects the game is over and restarts.The longer you survive, the more points you get, however the longer you are alive, the faster the objects spawn. The players score is rendered on screen and a local highscore board is displayed upon game over.



Results:

While our project was WebGL based instead of embedding it here we figured it was better experienced full screen as to better see oncoming objects and the player making it far more of a playable experience so it can be found on a separate hosted page at the link above (https://colechittim.github.io/final-480-chittim_daw/). Additionally by doing this it is easy to access and play on mobile devices as well.

Instructions for running and using the code:

The game can be run by simply opening the *index.html* file or visiting the page hosted at the link above (https://colechittim.github.io/final-480-chittim_daw/). If on a desktop the player can be controlled using the arrow keys or the WA(S)D keys and if on a mobile device, on screen buttons are displayed to control the player. Avoid oncoming asteroids for as long as possible to get a higher score.

Code guide:

index.html Main DOM for the website; contains framework for modification by the javascript
game/
The game folder contains the back end game driver code. In alphabetical order:

- **Cloud.js** Provides the cloud sprites for exhaust effect on the spaceship
- **Collision.js** Provides the math for enemy-player collision
- **ControlsManager.js** Adds all the event listeners for the game controls
- **Enemy.js** Defines the Enemy object with random spawn location and speed
- **GameManager.js** Describes the main game logic
- **GameUI.js** Hides/shows html elements depending on the game state

renderer/

The renderer folder contains all the code that drives the graphics from setup to asteroid. A lot of this code was heavily modified from Assignment 3.

- **glUtil.js** Main setup of canvas
- **loadobj.js** Generates info for the renderer from an obj file
- **mobileCheck.js** Big long regex expression to test for a mobile browser
- **renderer.js** GLSL source code for rendering the objects

css/

- **style.css** Contains html styling to make the website look all pretty