  
  
**Assignment Cover Sheet**

|  |  |
| --- | --- |
| **Subject Code:** | CSCI336 |
| **Subject Name:** | Interactive Computer Graphics |
| **Submission Type:** |  |
| **Assignment Title:** | Assignment 2 |
| **Student Name:** | Alphin Edgar D’cruz |
| **Student Number:** | 5561723 |
| **Student Phone/Mobile No.** | 056 - 4189634 |
| **Student E-mail:** | aedc999@uowmail.edu.au |
| **Lecturer Name:** | Hock Chuan Lim |
| **Due Date:** | 21 March 2018 |
| **Date Submitted:** | 22 March 2018 |

|  |  |
| --- | --- |
| **PLAGIARISM:** The penalty for deliberate plagiarism is FAILURE in the subject. Plagiarism is cheating by using the written ideas or submitted work of someone else. UOWD has a strong policy against plagiarism.  The University of Wollongong in Dubai also endorses a policy of non-discriminatory language practice and presentation.  **PLEASE NOTE:**STUDENTS MUST RETAIN A COPY OF ANY WORK SUBMITTED | **DECLARATION:** I/We certify that this is entirely my/our own work, except where I/we have given fully-documented references to the work of others, and that the material contained in this document has not previously been submitted for assessment in any formal course of study. I/we understand the definition and consequences of plagiarism.  **Signature of Student:** |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | **Optional Marks:** | | **Comments:** | |

https://my.uowdubai.ac.ae/images/scissors.gif

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Lecturer Assignment Receipt**(To be filled in by student and retained by Lecturer upon return of assignment) | | | **Subject:** | **Assignment Title:** | | **Student Name:** | **Student Number:** | | **Due Date:** | **Date Submitted:** | | **Signature of Student:** | | |

https://my.uowdubai.ac.ae/images/scissors.gif

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Student Assignment Receipt** (To be filled in and retained by Student upon submission of assignment) | | | **Subject:** | **Assignment Title:** | | **Student Name:** | **Student Number:** | | **Due Date:** | **Date Submitted:** | | **Signature of Lecturer** | | |

Table of Contents

[Introduction 3](#_Toc509439261)

[WebGL Code 3](#_Toc509439262)

[Common Files 3](#_Toc509439263)

[HTML File 3](#_Toc509439264)

[JavaScript File 3](#_Toc509439265)

[Function initialize 4](#_Toc509439266)

[Function startInteractiveSession 4](#_Toc509439267)

[vertMatrix File 4](#_Toc509439268)

[Data Structure: objectsToDraw 5](#_Toc509439269)

[CGI Modeling 5](#_Toc509439270)

[Subject Selection 5](#_Toc509439271)

[Modeling Fish from image 6](#_Toc509439272)

[Rendering Scene with webGL 6](#_Toc509439273)

[Conclusion 7](#_Toc509439274)

[References 7](#_Toc509439275)

Assignment 2 – Interactive Fish

# Introduction

The goal of this assignment was to produce a 3D interactive geometric computer graphic image (CGI) while developing on the fish theme from assignment one with a new model built from scratch. The interactive part of this assignment is 1. The spawning of fish on canvas click and 2. Interacting with an HTML slider to adjust the z axis rotation.

# WebGL Code

## HTML File

The html file is a simple script that contains the vertex with uniforms to adjust scales, transforms, spawns and view and fragment shader. All required java Script files are being included here.

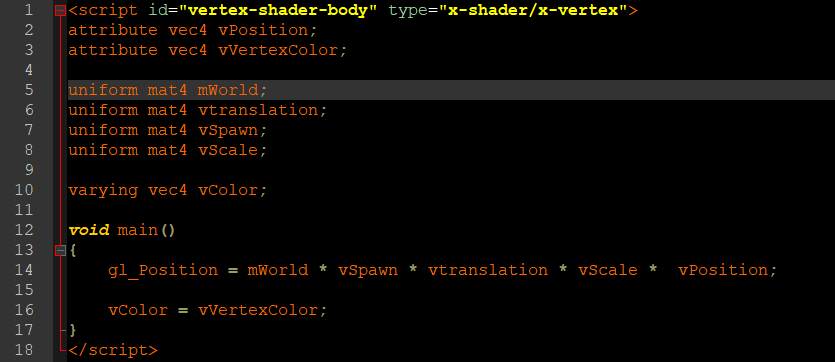


Figure 1 vector shader

## JavaScript File

The JavaScript file initializes webGL and contains the code to load shaders, loading data into GPU and rendering all divided into functions for ease of use.

### Function initialize

A simple function that will get canvas information from the html file, configure webGL and load the shaders.

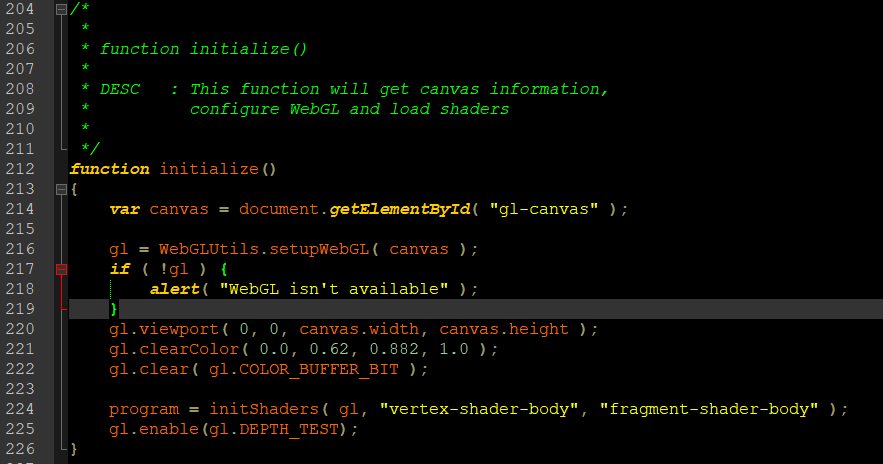


Figure 2: initialize function

### Function startInteractiveSession

The startInteractiveSession () function will enable the webGL interactive fish program, it contains the main render loop function drawScene () and functions that listen to mouse activity in order to enable interaction

#### Function drawScene

The drawScene function is the main render loop of the program and will go through every object to draw, sets the uniforms and draws them.

## vertMatrix File

The vertMatrix is another JavaScript file that only contains an array filled with vertex coordinate and color information. This file exists to improve readability of the code and data by separating them into separate files.

## Data Structure: objectsToDraw

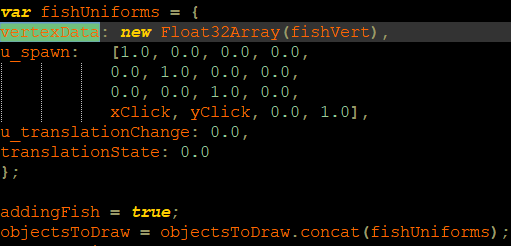
The main data structure of this program is objectsToDraw which is an array of objects with attributes: vertexData an array that holds the vertex position and color information, u\_spawn an array that holds the spawn matrix, u\_translateChange a variable that stores the rate of change of translation and u\_translateState a variable that stores the current x axis translation from the spawn of the object. The objectsToDraw array is looped through to draw and apply uniforms for each object.

Figure : objectsToDraw

# CGI Modeling

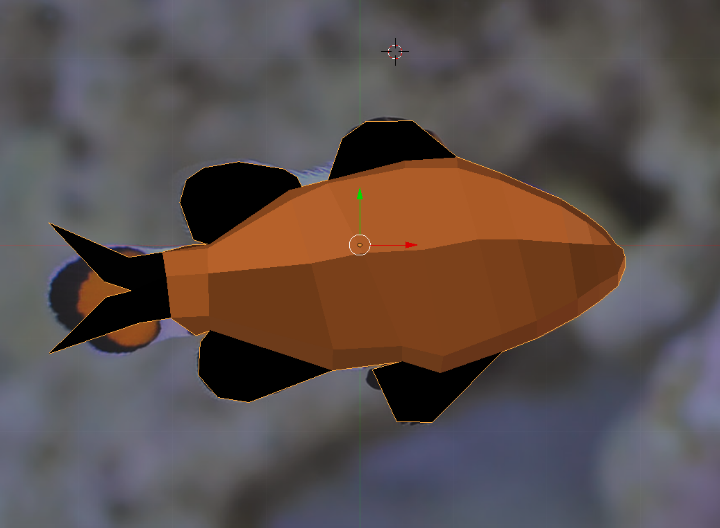
## Subject Selection

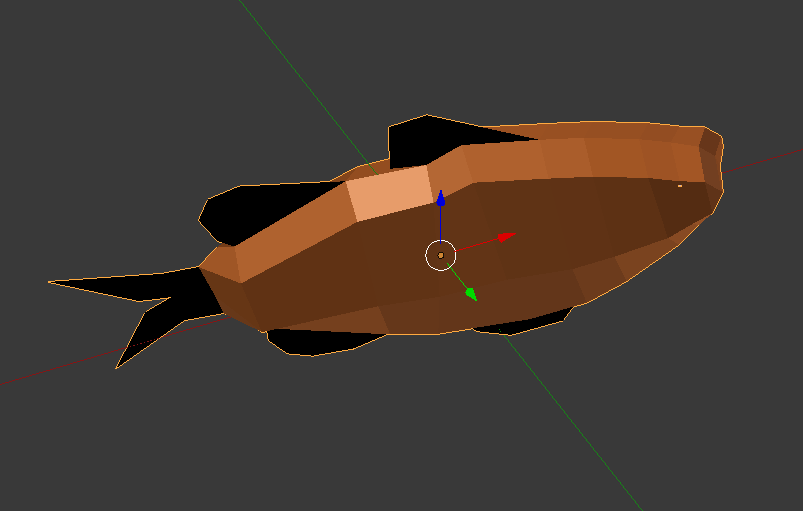
Just like assignment one an appropriate subject needed to be selected and this time I’ve opted for is a clown fish (as seen in Figure 4). Once again, a realistic fish model was the aim of this assignment and hence choosing a real fish as a model was only appropriate and the clown fish was chosen for its colorful nature.

Figure : Five things you know to know about types of aquarium fish

## Modeling Fish from image

With the subject now determined the next step was to model the fish and this was achieved through blender where with the image as reference, the model was created with quads. The fish was initially constructed on a 2D plane and with the help of extrusion and some slicing and transformation I was able to achieve one half of a 3D fish and since the fish is symmetric, creating the other half was as easy as mirroring on the z axis.





## Rendering Scene with webGL

Once the vertex coordinates and color information have been determined the was to be entered into an array. Once the canvas has been setup, a left mouse down will spawn a fish using the earlier mentioned array for vertex info and the position of the cursor is used to calculate a spawn offset. Moving the mouse while left mouse is down will cause the fish to follow the cursor and on left mouse up a constant x translation is added to the fish while maintaining the last spawn location.

# Conclusion

I’ve started this assignment with little knowledge of how animation, transformation and interaction work, the most challenging part was in understanding how to render different objects with different translations. Overcoming this hurdle taught me a lot about the importance of code and its has taught me about how to effectively use shaders to render our desired effects. In all this was a very insight full assignment that truly will help me in the future.

# References

Five things you know to know about types of aquarium fish [ONLINE]. Available at: http://www.the-pet-directory.com/five-things-you-need-to-know-about-types-of-aquarium-fish-today-types-of-aquarium-fish/ [Accessed 21 March 2018].