

Assignment 1

The evaluation of this assignment's tasks as well as the tasks of Exercise 3 and Exercise 4 will be held in the lab class **24th October** and corresponds to **4 points (20%)** of the total grade.

Tasks

The tasks of this assignment are:

1. Detect the car's collision with oranges, butters and road limits (the cheerios) by using surrounding boxes aligned with the axis (axis aligned bounding boxes - AABB). When the car reaches the limits of the track (collision with Cheerios) or a butter, it should stop, even if the user continues to press the button. When collided by the car, the butters or cheerios move slightly. If the car is hit by an orange, the player loses the race and the car should be placed at the start line.
2. Apply simultaneously at least two textures to the table. These textures should react to the lighting. Take the example of the multitexturing demo attached to this assignment, which uses texture maps in TGA format.
3. Allow to pause the game when the user presses a key ('S' key), showing a message while paused. Implement the counting mechanism of lives, points and the endgame. We suggest the following game play: at the beginning the player has five lives and lose a life each time the car is hit by an orange or falls off the table. Implement a HUD (Head-up Display) to show the remaining lives as well as the number of won points. When the player loses all lives, you should see a game over message and it should be possible to start a new game by pressing a button ('R' key).

Grading

- 1 – Graphic modelling **[2.5 points]**
- 2 – Cameras **[2.5 points]**
- 3 - Game elements' movement **[3 points]**
- 4 - Lighting of the scene **[5 points]**
 - a. Directional light **[1 point]**
 - b. Six point lights **[2.5 points]**
 - c. Spot light **[1.5 points]**
- 5 – Collision detection **[3 points]**
- 6 – Texture mapping **[2 points]**
- 7 – HUD (lives, score, reset, pause) **[1 point]**
- 7 – Graphic quality and gameplay **[1 point]**

Tips and Notes

1. To load images with other formats, you can use one of the following libraries:
 - **DevIL**: <http://www.lighthouse3d.com/cg-topics/code-samples/loading-an-image-and-creating-a-texture/>
 - Libpng: <http://www.libpng.org/pub/png/libpng.html>
 - Libjpg: <http://www.iijg.org/>
 - SOIL: <http://www.lonesock.net/soil.html>
 - GLI: <http://gli.g-truc.net/>
 - FreeImage: <http://freeimage.sourceforge.net/>
2. Displaying text messages to play and pause may be implemented through an application of a texture to an object. Remember that the use of freeglut's functions that manipulate strings, strokes and bitmaps are deprecated from the OpenGL version 3.1. So you must use other alternatives that can be found in the following links:

http://spacesimulator.net/tutorials/OpenGL_bitmapped_fonts_tutorial_3_3.html

<http://www.wildfiregames.com/forum/index.php?showtopic=17365>

<http://www.mbsoftworks.sk/index.php?page=tutorials&series=1&tutorial=12>

<http://www.lighthouse3d.com/very-simple-libs/vsfl/>

3. It is intended that the display information of points and lives is carried out through a Head-Up Display (HUD) which involves the use of a second orthogonal projection to the drawing. This projection will be independent of the active camera. In this case it may be worth doing push and pop the stack projection matrices.
4. If the implementation of the HUD is not possible it is suggested that students display that information to the header of the graphics window. In this case, students will be evaluated with 75% of the respective grade.