

Assignment #06 – Final Project

Demonstrations will be done: 11AM – 1PM, Tues 20th Nov 2018

Coursework %: approximately 65%

The purpose of this assignment is to allow you to extend the theory you learned in lectures by creating an appealing 3D object, character or vehicle.



Figure 1: example ideas for vehicle, object and creature, ranging from cartoon-rendering to realistic rendering. Complex hierarchy and model for the dragon, simplified geometry and hierarchy for the car. Focus can be on advanced rendering or animation, or both.

1. Outline

- This assignment is strictly **individual** (no groupwork).
- You are required to create an animated 3D object/character/vehicle using modern shader-based OpenGL (version 3 or higher). Specifications at the end.
- You will be required to **demonstrate your working program** to the demonstrators on the 20th November and they will grade you based on what you show them. The demo day should be treated like an exam, where attendance is compulsory. We will be using the timeslot from 11am – 1pm (ICT Lab 1 & 2) to do this. There will be no lecture on the 20th November.
- You will also be required to submit via Blackboard on 20th November:
 - a PDF report that explains your design and technical choices, with images.
 - all code source files and assets in a ZIP file on Blackboard
 - a youtube link to a short demonstration video-recording of your game (videos can be recorded using a tool such as FRAPS or Nvidia Shadowplay).

Try to demonstrate the technical capabilities of your project with visually pleasing images/videos.

- If you fail to show up for the lab or to submit your report on time, you will be reported as absent and will receive a grade of 0%.
 - You can use GLUT or any other library you like for systems calls, windowing, menus, interaction devices and other such tasks
 - You can use Windows, Linux or whatever platform you wish, as long as you can demonstrate your project in the lab on demo day
 - It is allowed to use a library to load models, as long as this is acknowledged. It is also allowed to use a library for some special effect, extra to the core functionality, such as physics, as long as this is acknowledged. If in doubt, ask me or demonstrators.
 - It is **not** allowed to use a graphics engine (e.g., UE4, Unity, etc.). This is a test of your ability to program the basic 3D graphics functionality covered in class, so no higher-level libraries or engines are allowed for rendering, camera transformations, etc.
 - Be sure to attend labs and ask the demonstrators for help.
 - Be aware that demonstrating code that was not created by you is considered **cheating** and will be reported as such.
-

2. Specifications:

Your object, vehicle or creature can be any kind you like, but the project must have the following basic functionality:

- 3-dimensional objects and views. (note: fixed top-down orthographic view not allowed).
- Consistent scene environment (teapot on table, car on road, etc.)
- User interaction and camera-control (**approx. 10%**)
 - a. user should be able to move around the scene using the keyboard and/or the mouse. At a minimum, implement moving forwards and backwards, turning left and turning right. Extra marks will be given for extra functionality (i.e., additional degrees of freedom of rotation of the camera).
- Implementation of the Phong Illumination model (**approx. 10%**)
 - a. 1 light source
 - b. Any type of light may be used (point, directional, spotlight)
 - c. Normal must be transformed correctly
 - d. Shading must use a combination of ambient, diffuse, and specular lighting
 - e. Any shading model may be used (flat, Gouraud, Phong) as appropriate for your scene rendering style
- At least one, well designed hierarchical animated object, creature or vehicle (no marks for a nonsensical hierarchies). Minimum requirement of a single one-to-one hierarchy. Extra marks for more complex hierarchies (e.g., one-to-many hierarchy of 4 wheels on a car, etc.). (**approx. 10%**)
- Texture mapping (**approx. 10%**)

This basic functionality is worth approximately 40% of the project mark but you must have made a solid attempt at some advanced features to obtain the full 40%.

The final 60% (approx.) will be given for advanced features, with approximately 15% per perfectly-executed feature – simple features will receive less, elaborate or well-designed features will receive more. You have some scope to put effort into your favourite aspects of 3D graphics, or experiment with new ideas.

Advanced Features can include the following, or indeed others that you think of:

- Advanced rendering/lighting effects
- Ray-tracing
- Advanced animation
- Non photorealistic rendering
- Collision detection
- Simulated dynamics or physics of any sort
- Intelligent Characters
- Great models that you made yourself
- Height-mapped terrain
- Procedurally generated terrain or meshes
- 3D “picking” to select game objects with the mouse cursor
- More advanced texturing effects; multi-texturing, environment-mapping, bump-mapping, specular-mapping, etc.
- Partially transparent geometry using alpha-blending
- Smoke or fog
- Crowds using boids or other
- On-screen control panels (using QT or other library)
- cube mapping for sky-boxes, environment mapping, or refraction
- ??? – your own feature ideas

Note: The [approximate] marking scheme provided shows the maximum marks that can be obtained for each section if completed perfectly. Merely attempting a section does not imply the full score indicated. You should aim to include bonus feature(s) and will be penalised if you have not attempted more than the basic functionality.

Some example ideas:

- A train using advanced shaders for metal, with smoke coming out of the chimney
- A toy character rendered using cartoon-shading, with cartoon-like animation
- A moving statue, using marble materials and shaders
- A metal teapot that can pour liquid
- A ship with billowing sails
- A fire-breathing dragon with flapping wings
- ??? – your own idea