

## File Loader for 3D Geometric Models

*Supervisor: Abel Gomes**Scribe: A. Gomes*

In this first project, the goal is to write a C++ program to load 3D meshes from files. The file type can be OFF, OBJ, PLY or else. The program must be an OpenGL application, preferably using modern OpenGL. The data structure to hold the mesh in memory is specified below per student.

## 1 Specific Learning Goals

After completing this project, students should know and be able to:

1. To deal with any geometric data structure for meshes.
2. To identify which geometric data structure behind each file type.
3. To eventually to convert the geometric data structure that sustains a given file type into the geometric data structure put forward by the instructor for each student.

## 2 Web Links

### OBJ format:

[https://en.wikipedia.org/wiki/Wavefront\\_.obj\\_file](https://en.wikipedia.org/wiki/Wavefront_.obj_file)  
<http://www.opengl-tutorial.org/beginners-tutorials/tutorial-7-model-loading/>  
<http://paulbourke.net/dataformats/obj/>  
<http://www.fileformat.info/format/wavefrontobj/egff.htm>  
<http://people.cs.clemson.edu/~dhouse/courses/405/docs/brief-obj-file-format.html>

### PLY format:

[https://en.wikipedia.org/wiki/PLY\\_\(file\\_format\)](https://en.wikipedia.org/wiki/PLY_(file_format))  
[http://www.cc.gatech.edu/projects/large\\_models/ply.html](http://www.cc.gatech.edu/projects/large_models/ply.html)  
<http://people.sc.fsu.edu/~jburkardt/data/ply/ply.html>  
<http://www.dcs.ed.ac.uk/teaching/cs4/www/graphics/Web/ply.html>

### OFF format:

[https://en.wikipedia.org/wiki/OFF\\_\(file\\_format\)](https://en.wikipedia.org/wiki/OFF_(file_format))  
[http://segeval.cs.princeton.edu/public/off\\_format.html](http://segeval.cs.princeton.edu/public/off_format.html)  
<http://www.geomview.org/docs/html/OFF.html>  
<http://people.sc.fsu.edu/~jburkardt/data/off/off.html>  
<http://www.holmes3d.net/graphics/offfiles/>

### 3 Project Steps

This project consists of the following steps:

1. To identify the geometric data structure associated to each type of geometry file (e.g., OBJ). Is it a soup of triangles? Is it an indexed facet-based data structure? Is it an edge-based data structure? Or else?
2. To code a function to read the geometric model from a specific-type file into the geometric data structure indicated by the instructor. This is the loader!
3. To code a function to write the geometric model into a specific-type file. This is an extra that gives 0.5 marks in the final grade of the course.
4. Email a .zip archive identified by student number (e.g., 12345project1TJV.zip) to `agomes@di.ubi.pt`. The project preferably should run from the command line, i.e., out of the environment of development of software. This means that you have to use a makefile. The deadline to deliver the project is on April 3 (Monday, 24:00). **Projects not working will not be evaluated. Projects delivered after the deadline will have a penalty of 0.5**

### 4 File Types to Students

Christian Lopes: OBJ + quad-edge data structure

Tiago Mendes: PLY + quad-edge data structure

José Manteigueiro: OFF + quad-edge data structure

António da Silva: OBJ + winged-edge data structure

Fábio Machado: PLY + winged-edge data structure

André Louro: OFF + winged-edge data structure

Miguel Rodrigues: OBJ + half-edge data structure

João Silva: PLY + half-edge data structure

Jusualdo Figueira: OFF + half-edge data structure

José Nunes: OBJ + cell-tuple data structure

Nanikafuako André: OFF + cell-tuple data structure

Leonel Antnio: OBJ + cell-tuple data structure

Willyan Dworak: PLY + Woo' symmetric data structure

Ezio Filho: OFF + Woo' symmetric data structure

João Caseiro: OBJ + Woo' symmetric data structure

### 5 Geometric Data Structures

See classes and search on Google.