

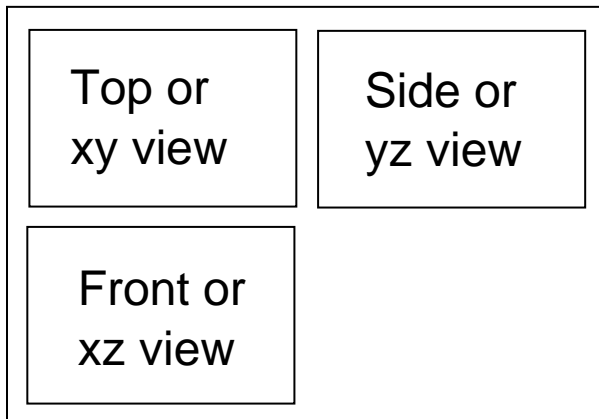
Assignment 1

CEG476/CEG676

Computer Graphics I

Assignment 1:

Please implement a computer program that is capable of rendering a scene that consists entirely of polygons using the direct drawing method of OpenGL. Your software should be able to generate three images of the scene defined by a PLY file using the following layout:



A description of the PLY file format can be found, for example, at the following web page:

<http://astronomy.swin.edu.au/~pbourke/dataformats/ply/>

Assignment 1

The PLY file format

The PLY file format uses a header containing a description of the format of the following data. Most commonly, the header defines a list of vertices and then a list of faces (which can be triangles, quads, or any other polygonal structure). After the header, the data entries follow immediately one item after the other.

So, usually the structure looks like this:

Header

Vertex List

Face List

(lists of other elements)

Assignment 1

PLY header

First, a PLY file has an entry to identify itself as such and then defines its format (ASCII, binary). We will use the ASCII format so that the first two lines of our PLY files look like this:

```
ply  
format ascii 1.0
```

Then, the following data items will be declared:

```
element vertex  
property float x  
property float y  
property float z
```

Values can be of type float, float32, uchar, int, or even vertex_indices (a list of vertex indices)

Assignment 1

Example (cube)

```
ply  
format ascii 1.0  
comment created by platoply  
element vertex 8  
property float32 x  
property float32 y  
property float32 z  
element face 6  
property list uint8 int32 vertex_indices  
end_header
```

Assignment 1

Example (data entries for a cube)

-1 -1 -1

1 -1 -1

1 1 -1

-1 1 -1

-1 -1 1

1 -1 1

1 1 1

-1 1 1

4 0 1 2 3

4 5 4 7 6

4 6 2 1 5

4 3 7 4 0

4 7 3 2 6

4 5 1 0 4

Assignment 1

Adding more information

You can add more properties to the data entries. For example, color information could be added to the vertices:

```
ply  
format ascii 1.0  
comment created by platoply  
element vertex 8  
property float32 x  
property float32 y  
property float32 z  
property red  
property green  
property blue
```

Assignment 1

Useful OpenGL functions:

```
void glOrtho (GLdouble left,  
             GLdouble right,  
             GLdouble bottom,  
             GLdouble top,  
             GLdouble near,  
             GLdouble far);
```

Specifies projection in OpenGL similar to 2-D example, therefore defining visible volume.

Assignment 1

Useful OpenGL functions:

```
void gluLookat (GLdouble eyex,  
                GLdouble eyey,  
                GLdouble eyez,  
                GLdouble centerx,  
                GLdouble centery,  
                GLdouble centerz,  
                GLdouble upx,  
                GLdouble upy,  
                GLdouble upz) ;
```

Specifies view in OpenGL looking from eye onto center, camera pointing up towards up vector.

Assignment 1

Useful OpenGL functions:

```
void glViewport (GLint x,  
                GLint y,  
                GLint width,  
                GLint height);
```

Specifies a viewport in OpenGL, i.e. a sub-area within the window so that drawing is only visible in that area.