Assignment Report

BM40A0600 Introduction to Computer Graphics

Iakov Lushin (0458163)

2016

# OpenGL Part

## General Description

The motivation for the scene is the Christmas and the New Year celebration. The blinking light represents the colored lights of a Christmas tree. Unfortunately, there is no Christmas tree itself.

## Program Execution

Only standard components are required for running the program.

Compilation Command (on Linux):

gcc -o assignment assignment.c shader.c matlib.c bmploader.c -lGL -lGLEW -lglut -lm

No initial information is required to run the program, just its execution.

No special steps are required after the program finishes.

## Program Controls

General controls are shown in the Table 1.

Table 1. General controls.

|  |  |
| --- | --- |
| **Button(s)** | **Functionality** |
| ESC/q | Exits the program. |

Light controls are presented in the Table 2. All lights off equals to full (brightest) ambient light.

Table 2. Light controls.

|  |  |
| --- | --- |
| **Button(s)** | **Functionality** |
| 0 | Toggles white (very dim) ambient light. |
| 1 | Toggles white (dim) directional light. |
| 2 | Toggles multicolored point light with specular reflections. |
| 3 | Toggles full (brightest) ambient light. |

Camera controls are presented in the Table 3.

Table 3. Camera controls.

|  |  |
| --- | --- |
| **Button(s)** | **Functionality** |
| x/X, y/Y, z/Z | Moves camera in negative/positive direction of x, y or z axis. |
| a/A, s/S, d/D | Moves point of look in negative/positive direction of x, y or z axis. |
| i/o | Zooms in/out. |
| n/N, f/F | Moves the nearest or farthest border of view towards/from the camera. |
| p | Toggles perspective/orthogonal projection. |

## Implemented Items

The list of implemented items can be found in the Table 4

Table 4. Items implemented in the OpenGL program.

|  |  |
| --- | --- |
| **Item** | **Points** |
| **User manual** | **1** |
| **Geometry of the model** | |
| *3D model* | |
| Number of geometric entities defined by the user: 3   * House cube (x3) * House roof (x3) * Ground plane (x1) | 2 |
| *Transformation matrices* | |
| Translation | 1 |
| Definition of the projection | |
| *3D projection (from 3D to 2D)* | |
| Location of the camera, focusing the camera | 1 |
| Projection | 1 |
| **Visible surfaces** | |
| *Z-buffer* | 1 |
| **Lighting of the model** | |
| *Ambient light* | 1 |
| *Directed light (not mentioned in the original table)* | 1 |
| *Point light source* | 1 |
| **Surface properties** | |
| *Colors (number of colors > 1): 2 colors (top and bottom parts of the house cubes)* | 1 |
| *Reflection of the light* | 1 |
| *Textures: roof and ground snow textures* | 2 |
| Interactive control | |
| *Changing projection (see controls)* | 1 |
| *Definition of the camera (see controls)* | |
| Location of the camera | 1 |
| Changing focus point | 1 |
| Optical properties of the camera: zooming | 1 |
| *Changing lighting (see controls)* | 1 |
| *Reading data from a file: reading bmp texture files* | 1 |
| *Some other interactive property: changing of the ground texture (change with 7, 8, 9)* | 1 |
| **Animation** | |
| *Time functions* | 1 |
| **GLSL programming** | |
| *Vertex shader* | 1 |
| *Fragment shader* | 1 |
| **Total** | **24** |

# POV-Ray

## General Description

The motivation for the scene is the room lit with moonlight.

## Program Execution

Only standard components are required for running the program.

Compilation Command (on Linux):

povray assignment.ini +Q9 +W1000 +H800

No initial information is required to run the program, just its execution.

No special steps are required after the program finishes.

## Implemented Items

The list of implemented items can be found in the Table 5.

Table 5. Items implemented in the POV-Ray program.

|  |  |
| --- | --- |
| **Item** | **Points** |
| **User manual** | **1** |
| **Geometry of the model** | |
| Number of different 3D objects: 5   * Room box * Table * Flashlight * Laptop * Picture on the wall | 1 |
| **Definition of the projection** | |
| Camera model | 1 |
| **Lighting** | |
| Directed light in infinity (moonlight) | 1 |
| Point light source | 1 |
| Spot light (flashlight) | 1 |
| Area light (laptop screen) | 1 |
| **Properties of the geometric entities** | |
| Predefined textures (table) | 1 |
| Image mapping (laptop screen) | 1 |
| Bump mapping (picture on the wall) | 1 |
| **Algorithms** | |
| Animation with time functions (moving flashlight) | 1 |
| **Some other program feature** | |
| Extensive usage of Constructive Solid Geometry | 1 |
| Object complexity | 1 |
| **Total** | **13** |