1.8 Objectives: Assignment 3

Due: Wednesday, June 17th [Week 7].

Name: _____

UserID: _____

Student ID: _____

___ **1:** `a3mark.lua` runs and displays correctly.

___ **2:** The puppet's proportions are reasonable and the spheres are joined together in a logical fashion. The spheres are "hinged" to their neighbours at their ends—they do not rotate about their centres.

___ **3:** Picking works correctly and reliably.

___ **4:** Selection records are kept correctly so that any sequence of picks-to-select and picks-to-unselect works. Multiple selections are supported. Selected portions of the puppet are clearly indicated—e.g. selected parts change colour, or their edges are highlighted.

___ **5:** The puppet can be globally rotated and translated for viewing purposes. The rotation user interface is a virtual trackball. A circle representing the virtual sphere can be turned on or off from the menu. It must be clearly visible when it is turned on. The puppet's configuration can be reset from the menu.

___ **6:** A well-designed hierarchical data structure is employed. Each sphere is drawn as an instance of a single vertex buffer object sphere.

___ **7:** The joint movements are correct and the angles are restricted so that no grossly unnatural configurations are allowed. The puppet does not fly apart or distort during any sequences or combinations of actions.

___ **8:** Z-buffer, backfacing and frontfacing polygon hidden surface removal are implemented and each can be toggled on and off.

___ **9:** The puppet is lit such that its 3D structure is clearly visible.

___ **10:** An undo/redo stack is maintained.

Declaration:

I have read the statements regarding cheating in the CS488/688 course handouts. I affirm with my signature that I have worked out my own solution to this assignment, and the code I am handing in is my own.

Signature:

CS488/688 S15 Copyright permission for Assignment 3

Check one.

☐ I grant permission to use the images of my Assignment 3 in the course webpage and/or t-shirt. I realize that I can revoke this permission for use in the webpage at any time by requesting so in writing or by email.

☐ I do NOT grant permission to use the images of my Assignment 3 in the course webpage or t-shirt.

Name (printed):

Student id:

User id:

Signature:

Date:

William McDonald
20418145
wmcdonal

CS 488 Assignment 2

Manual

The data structure is very similar to the original. Only a pointer to the node's parent has been added, so that when picking a geometry node we can find the corresponding joint parent. Several methods have been added to the Nodes to help.

Sphere Drawing

A single sphere vertex buffer was computed and is used for every sphere. The sphere is created by tilling the surface with quadrilaterals, which are then split in half to become triangles. There is also a single buffer for the normals of the sphere vertices which is used for the lighting computations of every sphere.

Picking

Picking was implemented by unique-colour picking. The scene is rendered to a QOpenGLFrameBufferObject with depth checking enabled and dithering disabled. The pixel that was clicked is retrieved and an id is computed from its RGB value. The joint corresponding to that id is found (if it exists) and added to a set of picked nodes.

Rotating Joints

There are a few components for rotating joints. Firstly, there is a jointMap which is a mapping of ids to JointNodes. This is to allow us to check for and avoid over-rotations.

Next, there is the undo/redo stack, which is a snapshot of the current state of rotation for each joint. There is finally the opMap, which is a map containing the net rotations per-id. The combination of these allows us to rotate multiple joints, and undo these rotations even when some joints are over-rotated.

When a rotation drag begins, a new entry is added to the top of the undo/redo stack. This top entry is modified in-place as the drag continues.

Structure of Puppet

Several abstractions were made in order to model the puppet. Firstly, there is a concept of a “drawable node”, which is simply a node with a child, and that child is the geometry node. Next there is the concept of a “limb”, which is a series of drawable nodes connected by joints. The puppet is constructed out of four limbs, and additional drawable nodes for the head, nose, shoulders and neck.

The head has a single joint which can rotate about Y and X. All other joints are restricted to rotating about X, as per the assignment specifications. There is nothing special in the code that would restrict other joints from rotating about the Y axis as well.

Assumptions

It is assumed that this program will be run on Ubuntu 14.04 with Qt version 5.2.1.

sum is: /usr/bin/sum

A3:
total 2996
129680442 drwxrwxr-x 3 wmcdonal wmcdonald 4096 Jun 17 03:04 ./
91491409 drwxrwxr-x 2 wmcdonal wmcdonald 4096 Jun 17 03:03 src/
129680451 drwxrwx--- 7 wmcdonal cs488 4096 Jun 17 02:59 ../
100437577 -rw-r--r-- 1 wmcdonal wmcdonald 526 Jun 17 02:57 sphere.vert
75242228 -rw-r--r-- 1 wmcdonal wmcdonald 911 Jun 17 02:57 sphere.frag
29282827 -rw-r--r-- 1 wmcdonal wmcdonald 3880 Jun 17 02:57 puppet.lua
29282821 -rwxr-xr-x 1 wmcdonal wmcdonald 2998025 Jun 17 02:57 puppeteer*
29282825 -rw-r--r-- 1 wmcdonal wmcdonald 23699 Jun 17 02:56 screenshot01.png
29282817 -rwxr-xr-- 1 wmcdonal wmcdonald 413 Jun 17 02:56 README*

A3/src:
total 116
129680442 drwxrwxr-x 3 wmcdonal wmcdonald 4096 Jun 17 03:04 ../
91491409 drwxrwxr-x 2 wmcdonal wmcdonald 4096 Jun 17 03:03 ./
126503304 -rw-r--r-- 1 wmcdonal wmcdonald 17352 Jun 17 03:02 Viewer.cpp
126503303 -rw-r--r-- 1 wmcdonal wmcdonald 144 Jun 17 03:02 scene_lua.hpp
126503315 -rw-r--r-- 1 wmcdonal wmcdonald 11752 Jun 17 03:02 scene_lua.cpp
126503314 -rw-r--r-- 1 wmcdonal wmcdonald 2590 Jun 17 03:02 scene.hpp
126503313 -rw-r--r-- 1 wmcdonal wmcdonald 3637 Jun 17 03:02 scene.cpp
126503312 -rw-r--r-- 1 wmcdonal wmcdonald 340 Jun 17 03:02 primitive.hpp
126503311 -rw-r--r-- 1 wmcdonal wmcdonald 314 Jun 17 03:02 primitive.cpp
126503310 -rw-r--r-- 1 wmcdonal wmcdonald 474 Jun 17 03:02 material.hpp
126503309 -rw-r--r-- 1 wmcdonal wmcdonald 457 Jun 17 03:02 material.cpp
126503308 -rw-r--r-- 1 wmcdonal wmcdonald 909 Jun 17 03:02 main.cpp
126503307 -rw-r--r-- 1 wmcdonal wmcdonald 208 Jun 17 03:02 lua488.hpp
116390179 -rw-r--r-- 1 wmcdonal wmcdonald 526 Jun 17 02:56 sphere.vert
31731407 -rw-r--r-- 1 wmcdonal wmcdonald 911 Jun 17 02:56 sphere.frag
31731405 -rw-r--r-- 1 wmcdonal wmcdonald 721 Jun 17 02:56 puppeteer.pro
31731403 -rw-r--r-- 1 wmcdonal wmcdonald 3880 Jun 17 02:56 puppet.lua
992039 -rw-r--r-- 1 wmcdonal wmcdonald 1031 Jun 17 02:56 a3mark.lua
91491423 -rw-r--r-- 1 wmcdonal wmcdonald 4117 Jun 17 02:56 Viewer.hpp
91491412 -rw-r--r-- 1 wmcdonal wmcdonald 1086 Jun 17 02:56 AppWindow.hpp
91491410 -rw-r--r-- 1 wmcdonal wmcdonald 4435 Jun 17 02:56 AppWindow.cpp

A3
A3/puppeteer 30196 2928
A3/puppet.lua 62736 4
A3/README 27971 1
A3/screenshot01.png 49923 24
A3/sphere.frag 30112 1
A3/sphere.vert 64540 1
A3/src
A3/src/a3mark.lua 31842 2
A3/src/AppWindow.cpp 11651 5
A3/src/AppWindow.hpp 64382 2
A3/src/lua488.hpp 60267 1
A3/src/main.cpp 45927 1
A3/src/material.cpp 62170 1
A3/src/material.hpp 28936 1
A3/src/primitive.cpp 33211 1
A3/src/primitive.hpp 64759 1
A3/src/puppeteer.pro 64906 1
A3/src/puppet.lua 62736 4
A3/src/scene.cpp 00364 4

A3/src/scene.hpp	38906	3
A3/src/scene_lua.cpp	38536	12
A3/src/scene_lua.hpp	34532	1
A3/src/sphere.frag	30112	1
A3/src/sphere.vert	64540	1
A3/src/Viewer.cpp	24862	17
A3/src/Viewer.hpp	10134	5