CT4510: Computer Graphics

PA #2

BOCHANG MOON

Programming Assignment #2

- Problem specification (10 points)
  - Change the title bar with your student ID (1 point)
  - Load two models (teapot and armadillo) and properly arrange them using modeling transformation (2 points)
    - Your screen should visualize the two models
  - Picking and selection
    - A. When a user is pressing the left mouse button on an object, print the following messages in your command window. (2 points)
      - e.g. “Teapot is selected” or “Armadillo is selected”
    - B. When a user is pressing the left mouse button on an object, you should visualize a specific color assigned to each object. (3 points)
      - e.g., red for Armadillo and green color for Teapot
    - C. When a user is not pressing the left mouse button on an object, you should do the original rendering (2 points) – this will be evaluated only when you implement B.
    - NOTE: if you use the mentioned naïve ways discussed in the class, no score will be given for the picking and selection.

Programming Assignment #2

- Submission:
  - Due date: 23:59:59, Friday, April 6th , 2018 (KST)
  - A zipped file with (file name should be “PA2_your student number_your name.zip”)
    - Your source code (a zipped file only with .h and .cpp)
    - A binary file (NOTE: change the file extension, e.g., XXX.exe -> XXX.dat)
      - Your binary file will be checked on a windows system.
    - Not any virus files (your final grade will be “F”)
    - TA email address: ta.cg.gist@gmail.com

** Make sure the binary file is working when it is opened.

- It means there should be no needs to do an extra job to run your program. (e.g., debugging, build, etc.)

Some Hints for PA #2

- Check the reference binary

Some Hints for PA #2

- OpenGL screen coordinates are different from Windows coordinates.
  - The y coordinate is opposite.
- void mouseButton(int button, int state, int x, int y) {
  - if (button == GLUT_LEFT_BUTTON) {
    - if (state == GLUT_DOWN) {
      - glReadBuffer(GL_BACK);
      - unsigned char pixel[3];
      - glReadPixels(x, g_height - y, 1, 1, GL_RGB, GL_UNSIGNED_BYTE, pixel);
      - // write down rest of your code to compare the pixel value (0 – 255) with the specified color
      - // print some text on your command window using the c function “printf(“XXX”);”
      - // store the model ID to a global variable so that it can be used in the draw functions.
    - }
  - }
  - glutPostRedisplay();
- }

Some Hints for PA #2

- As a simple way, you can draw two buffers (front and back buffers)

- void drawTeapot() {
  - …
  - // insert some code here
  - //
  - glDrawBuffer(GL_FRONT);
  - glCallList(g_teapotID);

  - // insert your code here
  - //
  - glDrawBuffer(GL_BACK);
  - glCallList(g_teapotID);
- }

Some Hints for PA #2

- As a simple way, you can write a new function for Armadillo model.
- void drawArmadillo() {
  - // this function will be very similar to the drawTeapot function
- }
- void display() {
  - …
  - // Draw your objects here
  - drawTeapot();// Draw a teapot
  - drawArmadillo();// Draw the armadillo
- }
- void main(int argc, char* argv[]) {
  - glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
- }