

# Assignment #2



# Purpose of Assignment

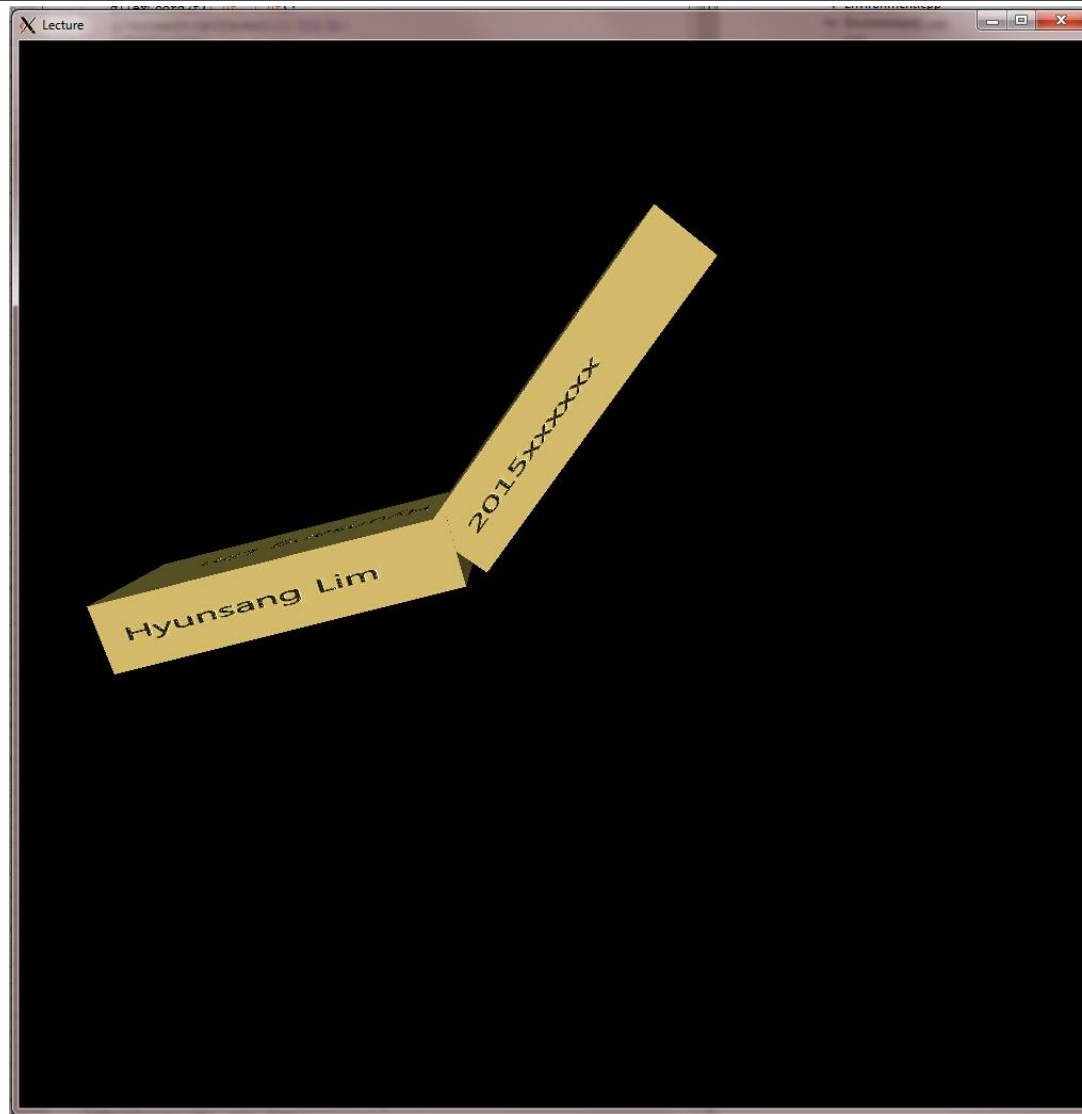
---

## Make Robot-arm Program with Shader

### ■ Requirements

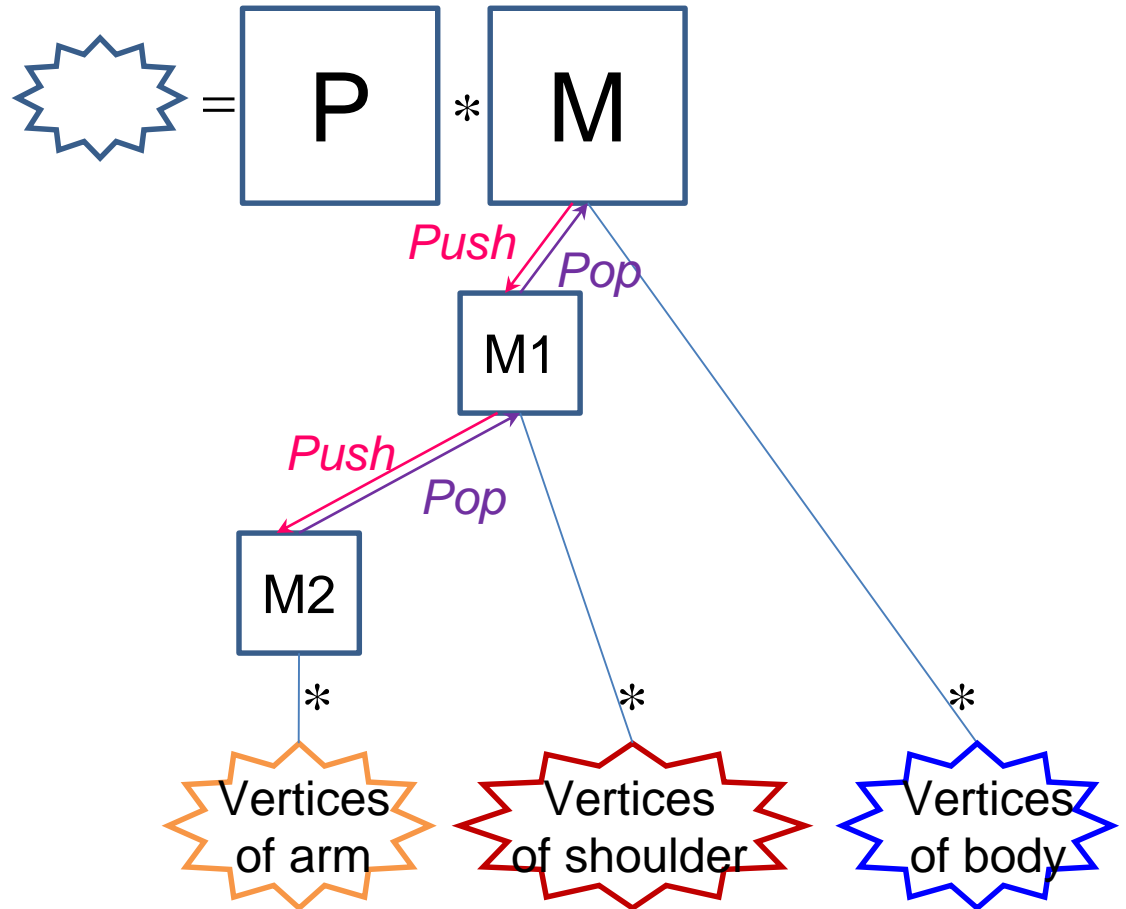
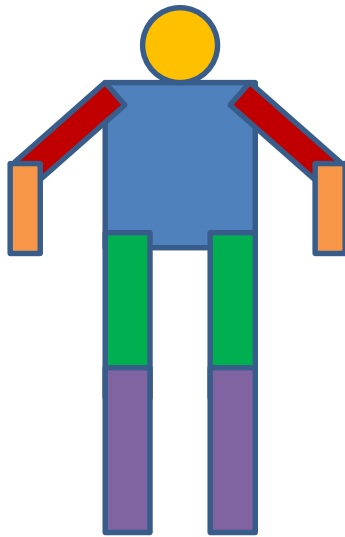
1. Texture Mapping
  - Texture Mapping(name, student\_ID)
  - Normal Mapping
2. Phong Lighting Model
  - Ambient, Diffuse, Specular Light with Phong Shading
3. Using Shaders(Vertex, Fragment)
4. Run at GPU server

# Result Example



# Push & Pop

We can manage the hierarchy by *glPushMatrix()*, *glPopMatrix()*.



# Transform Example : Robot Arm

```
int shoulder = 0, elbow = 0;
```

```
void display() {
```

```
    /*Initialize Drawing*/
```

```
    glPushMatrix();
```

```
        glRotatef(20, 1, 0, 1);
```

```
        glPushMatrix();
```

```
            glTranslatef(-1.0, 0.0, 0.0);
```

```
            glRotatef(shoulder, 0.0, 0.0, 1.0);
```

```
            glTranslatef(1.0, 0.0, 0.0);
```

```
        glPushMatrix();
```

```
            glScalef(2.0, 0.4, 1.0);
```

```
            glColor3f(1,0,0);
```

```
            glutSolidCube(1.0);
```

```
        glPopMatrix();
```

```
        glTranslatef(1.0, 0.0, 0.0);
```

```
        glRotatef(elbow, 0.0, 0.0, 1.0);
```

```
        glTranslatef(1.0, 0.0, 0.0);
```

```
    glPushMatrix();
```

```
        glScalef(2.0, 0.4, 1.0);
```

```
        glColor3f(1,1,0);
```

```
        glutSolidCube(1.0);
```

```
    glPopMatrix();
```

```
    glPopMatrix();
```

```
    glXSwapBuffers(dpy, win);
```

```
}
```

```
void keyPressEvent(char* key_string){
```

```
    if(strncmp(key_string, "Up", 2) == 0){
```

```
        shoulder = (shoulder+5)%360;
```

```
    }else if(strncmp(key_string, "Down", 4) == 0){
```

```
        shoulder = (shoulder-5)%360;
```

```
    }else if(strncmp(key_string, "Right", 5) == 0){
```

```
        elbow = (elbow+5)%360;
```

```
    }else if(strncmp(key_string, "Left", 4) == 0){
```

```
        elbow = (elbow-5)%360;
```

```
    }
```

```
}
```

```
int main(int argc, char *argv[]) {
```

```
    /*CreateWindow*/
```

```
    XEvent xev;
```

```
    while(1) {
```

```
        display();
```

```
        XNextEvent(dpy, &xev);
```

```
        if(xev.type == KeyPress){
```

```
            char *key_string = XKeysymToString(
```

```
                XkbKeycodeToKeysym(dpy, xev.xkey.keycode, 0, 0));
```

```
            keyPressEvent(key_string);
```

```
        }
```

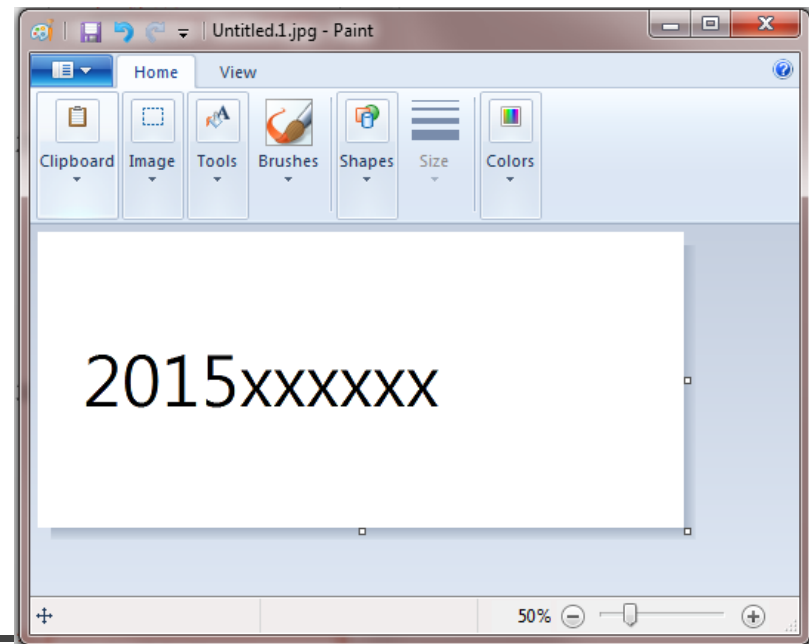
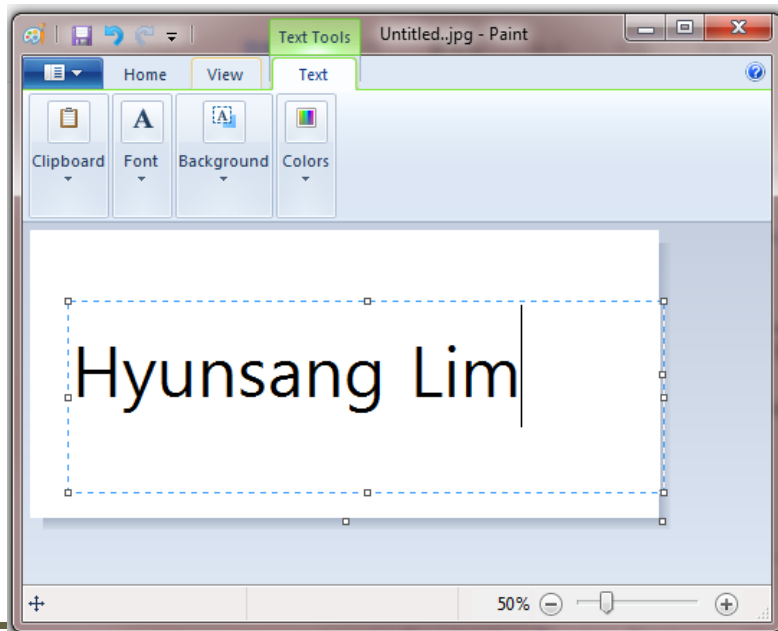
```
    }
```

```
}
```

# Make Your Texture Images

**Make two color image with your “Paint” program.**

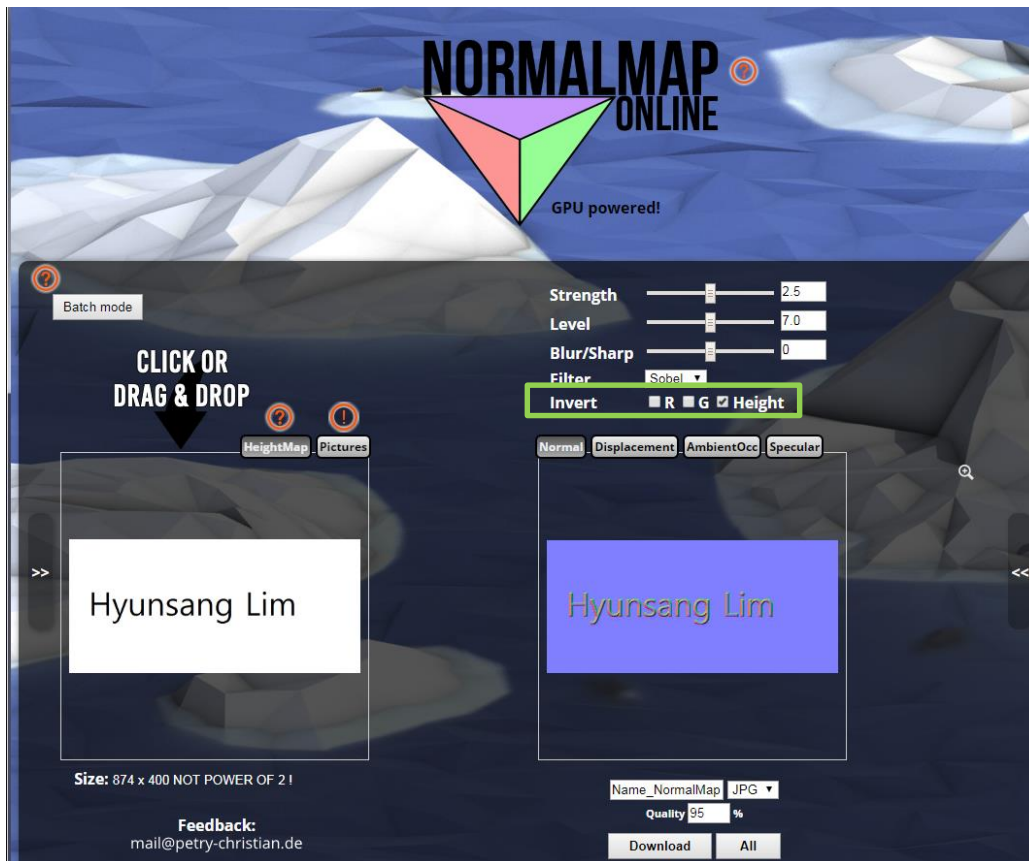
- Your name
- Your Student ID
- We recommend 800\*400 size



# Make normal image

You can convert Color image to normal map

■ <http://cpetry.github.io/NormalMap-Online/>



# Texture Images Example

**Transfer Image to your project folder on GPU Server**

- We'll give file transferring guide

Hyunsang Lim

2015xxxxxx

Hyunsang Lim

2015xxxxxx



# Programming-Hint: Draw Box

**glutSolidCube** function has no texture coordinates.  
You should use following modified function.

```
static void drawBox(GLfloat size)
{
    /*initialize vertex & normal value*/
    for (i = 5; i >= 0; i--) {
        glBegin(GL_QUADS);
        glNormal3fv(&n[i][0]);
        glColor3f(0.0f, 0.0f, 0.0f);
        glVertex3fv(&v[faces[i][0]][0]);
        glColor3f(1.0f, 0.0f, 0.0f);
        glVertex3fv(&v[faces[i][1]][0]);
        glColor3f(1.0f, 1.0f, 0.0f);
        glVertex3fv(&v[faces[i][2]][0]);
        glColor3f(0.0f, 1.0f, 0.0f);
        glVertex3fv(&v[faces[i][3]][0]);
        glEnd();
    }
}
```

# Programming-Hint: TBN Transform

With this **TBN matrix**, we can transform normals (extracted from the texture) into model space

```
//fragment Shader
varying normal; //normal from vertex shader
void main(){
    /*.....*/

    vec3 n = normalize(normal);
    vec3 b = normalize(vec3(0,0,1));
    vec3 t = normalize(cross(b, n));
    b = cross(t, n);
    mat3 TBNMatrix = transpose(mat3(t,b,n));

    vec3 normalFromMap; /*read from normapmap*/
    n = normalize(normalFromMap*TBNMatrix);

    /*.....*/
}
```

# Submit the Assignment

## Submit the zip file @ Blackboard

- File name must be “Assignment2\_StudentID\_Name.zip”

Ex. Assignment2\_2015000000\_박지혁.zip

- Must include

Src file

- c/c++ and header files
- Shader files

4 texture images

- 2 color
- 2 normal

Result running Image file

- Due date: November 4<sup>th</sup>

# **Guide: File Transfer to Server**



# FileZilla

## What is FileZilla

- FileZilla is the open source cross platform FTP software developed by Tim Kosse

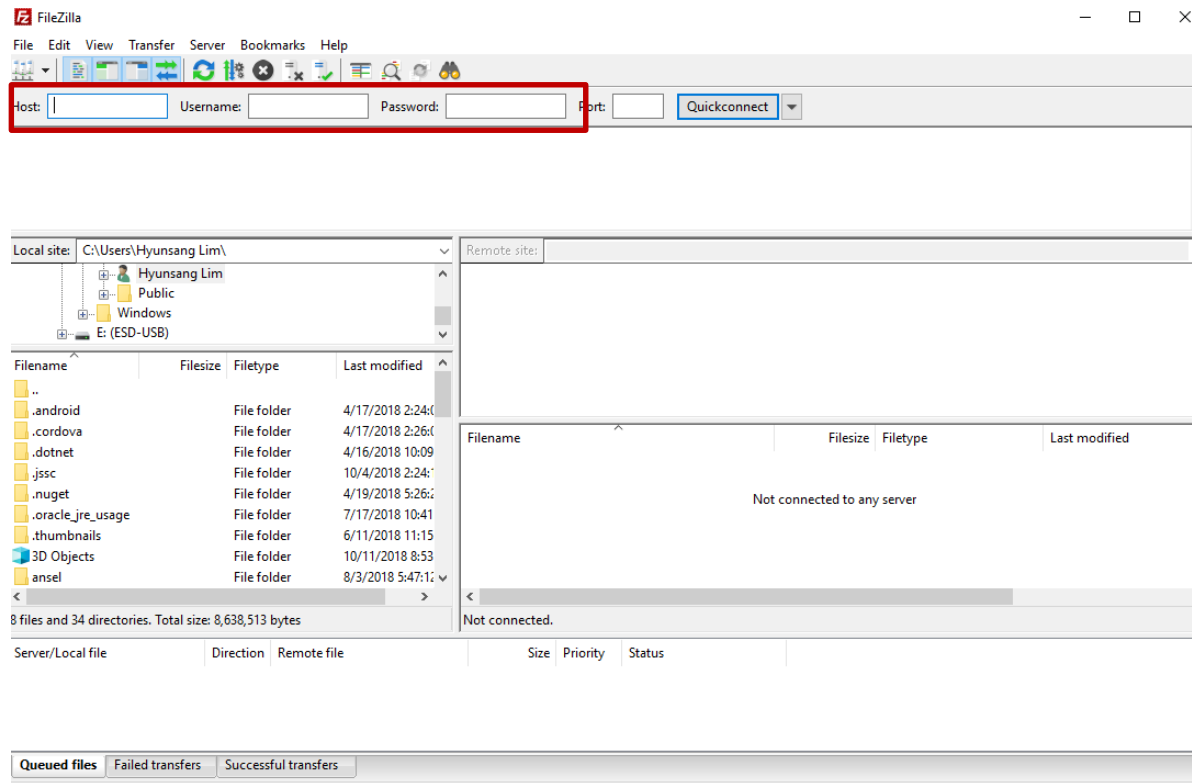
## Download link

- <https://filezilla-project.org/download.php?type=client>
- Recommending to Download with default settings



# FileZilla Connecting

Type 'sftp://163.152.20.246' at Host  
Type Username and Password



# FileZilla File Transfer

Transferring file by drag & drop file from to ③ ⑤

- ① Representing working states
- ② Representing folder tree of your PC
- ③ Representing sub-folders & files in selected folder of you PC
- ④ Representing folder tree of server
- ⑤ Representing sub-folders & files in selected folder of server

