Homework 1

COSE331 Computer Graphics

Goal

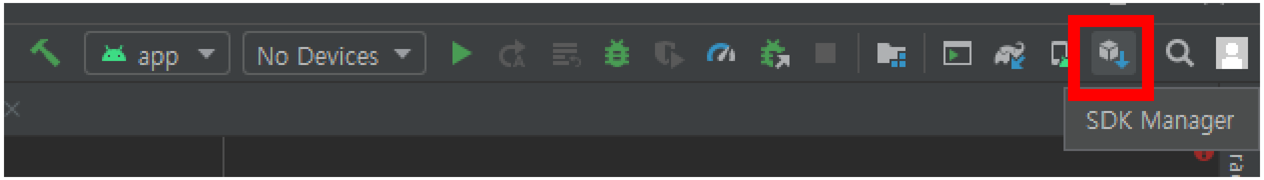
- Setting up Android Studio development environment
- Filling in some lines in the vertex shader
  - Vertex position
  - Vertex normal
  - Texture coordinates
- Filling in lines in the functions of the scene class in GL program

Android Studio

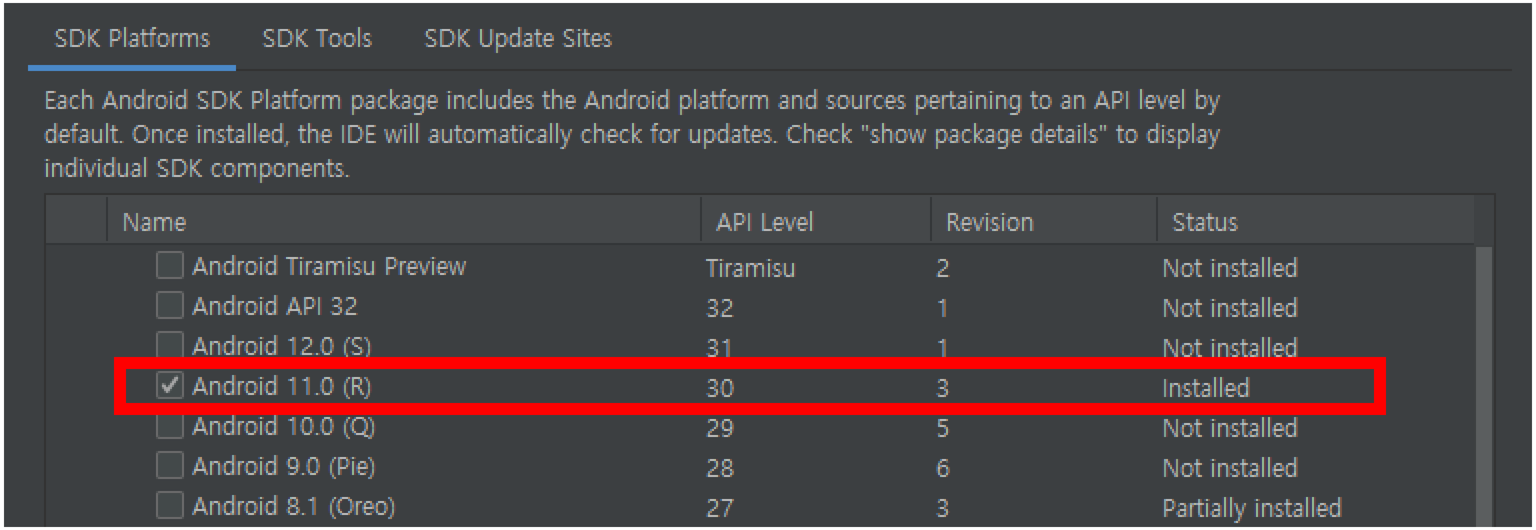
- **Android Studio** is the official integrated development environment (IDE) for the Android platform.
- Android Studio can be downloaded from the official website. [link]
- Recommended version : Android Studio Flamingo | 2022.2.1 Patch 2

Android SDK

- **Android SDK** can be installed through the SDK Manager in Android Studio.



- Use SDK **API level 30**



SDK Platforms | SDK Tools | SDK Update Sites

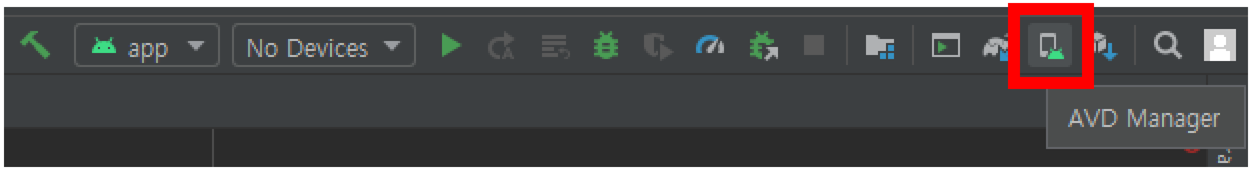
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, the IDE will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
□ Android Tiramisu Preview	Tiramisu	2	Not installed
□ Android API 32	32	1	Not installed
□ Android 12.0 (S)	31	1	Not installed
☑ Android 11.0 (R)	30	3	Installed
□ Android 10.0 (Q)	29	5	Not installed
□ Android 9.0 (Pie)	28	6	Not installed
□ Android 8.1 (Oreo)	27	3	Partially installed

Android SDK

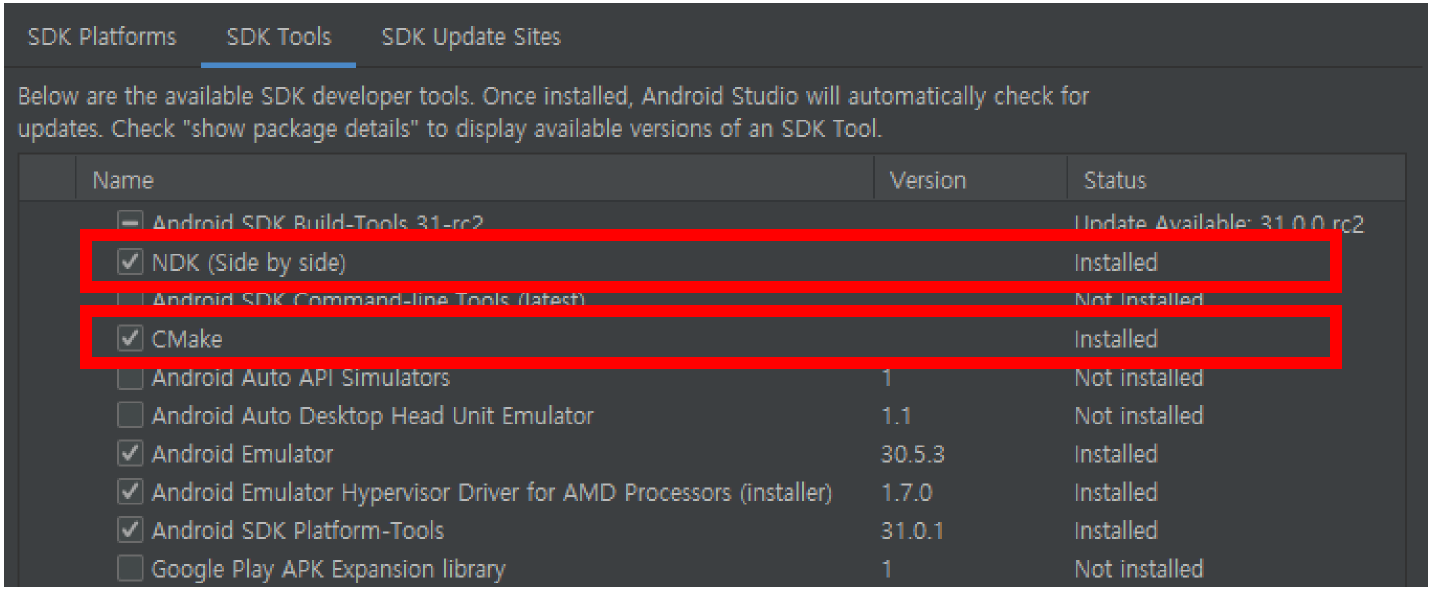
- You may need to install **additional platforms** to try USB debugging on your smartphone.

- Or you can use **Android Virtual Device (AVD)**.
  - To use AVD, you need to enable **hardware virtualization technology** (shown as **VT-x** or **SVM**) in BIOS settings.



Android SDK

- To use **C++ native language** on Android, you need to install the following tools.
  - CMake
  - NDK **(Use version 19.2.5345600)**

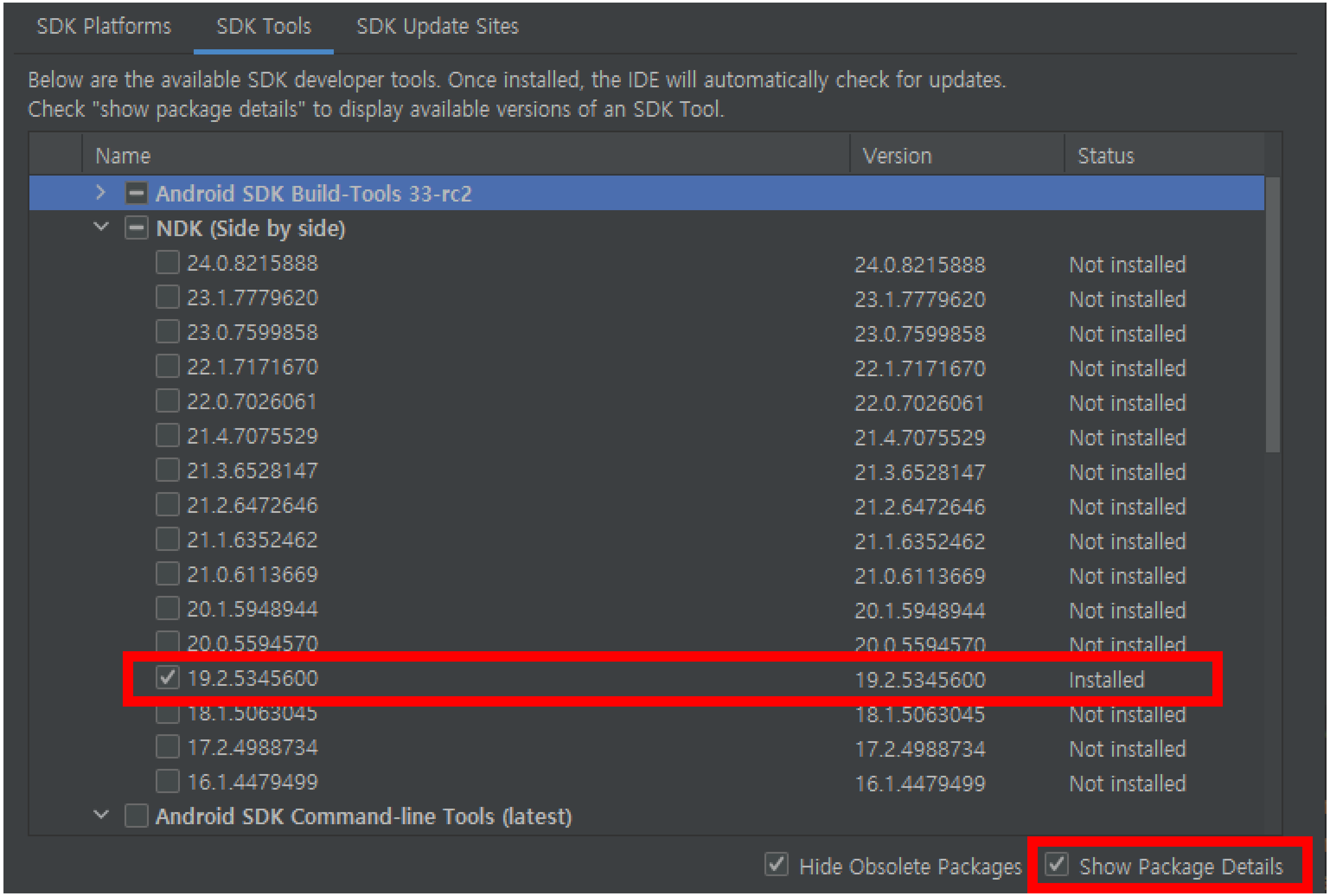


Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
▭ Android SDK Build-Tools 31-rc2		Update Available: 31.0.0 rc2
☑ NDK (Side by side)		Installed
□ Android SDK Command-line Tools (latest)		Not Installed
☑ CMake		Installed
□ Android Auto API Simulators	1	Not installed
□ Android Auto Desktop Head Unit Emulator	1.1	Not installed
☑ Android Emulator	30.5.3	Installed
☑ Android Emulator Hypervisor Driver for AMD Processors (installer)	1.7.0	Installed
☑ Android SDK Platform-Tools	31.0.1	Installed
□ Google Play APK Expansion library	1	Not installed

Android SDK

- Use **NDK version 19.2.5345600**.



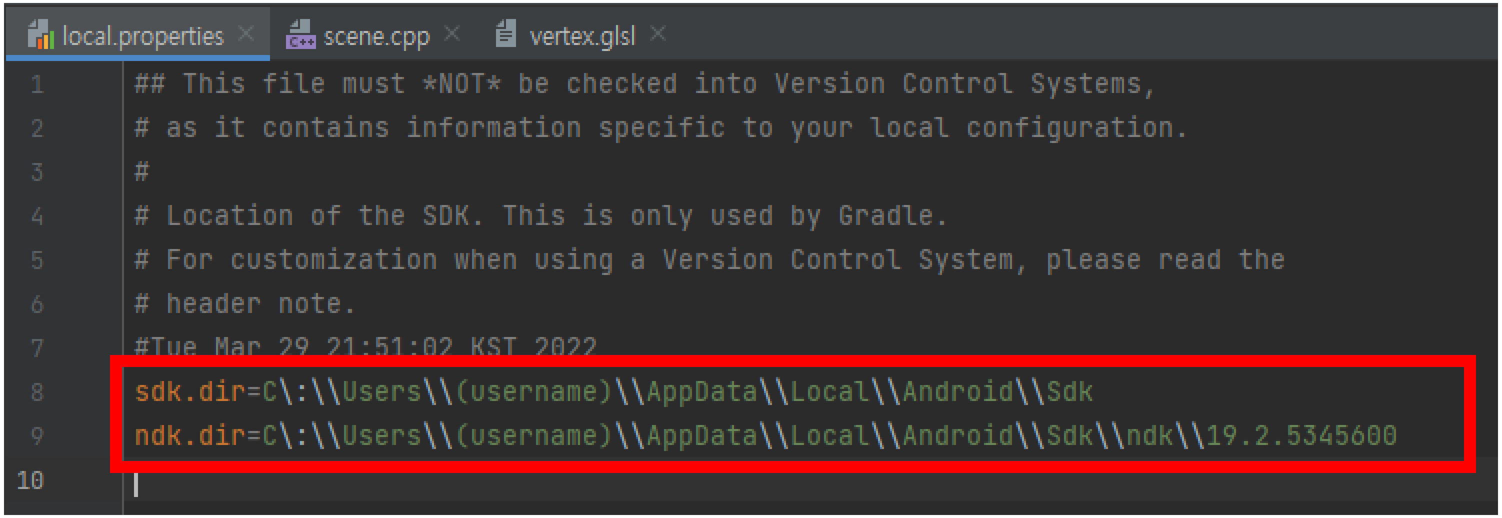
Below are the available SDK developer tools. Once installed, the IDE will automatically check for updates.
Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
> ▣ Android SDK Build-Tools 33-rc2		
⌄ ▣ NDK (Side by side)		
□ 24.0.8215888	24.0.8215888	Not installed
□ 23.1.7779620	23.1.7779620	Not installed
□ 23.0.7599858	23.0.7599858	Not installed
□ 22.1.7171670	22.1.7171670	Not installed
□ 22.0.7026061	22.0.7026061	Not installed
□ 21.4.7075529	21.4.7075529	Not installed
□ 21.3.6528147	21.3.6528147	Not installed
□ 21.2.6472646	21.2.6472646	Not installed
□ 21.1.6352462	21.1.6352462	Not installed
□ 21.0.6113669	21.0.6113669	Not installed
□ 20.1.5948944	20.1.5948944	Not installed
□ 20.0.5594570	20.0.5594570	Not installed
☑ 19.2.5345600	19.2.5345600	Installed
□ 18.1.5063045	18.1.5063045	Not installed
□ 17.2.4988734	17.2.4988734	Not installed
□ 16.1.4479499	16.1.4479499	Not installed
⌄ □ Android SDK Command-line Tools (latest)		

☑ Hide Obsolete Packages ☑ Show Package Details

Gradle Sync

- Before doing homework, you have to modify `local.properties` file.
  - Change (username) of `ndk.dir` and `sdk.dir` to **your PC username**.



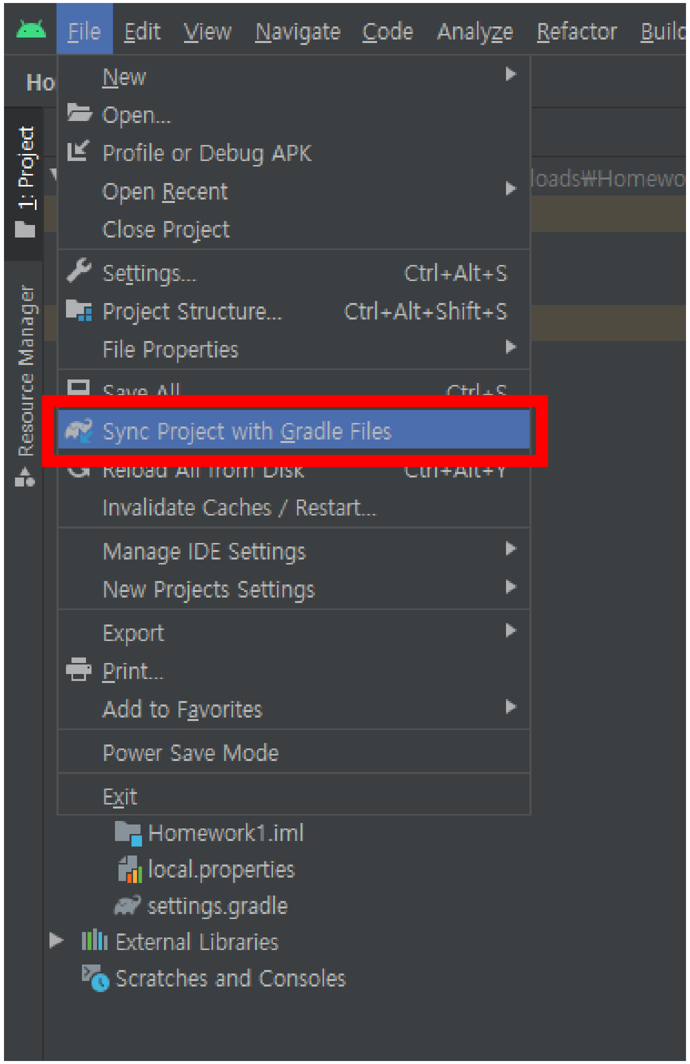
```
## This file must *NOT* be checked into Version Control Systems,
# as it contains information specific to your local configuration.
#
# Location of the SDK. This is only used by Gradle.
# For customization when using a Version Control System, please read the
# header note.
#Tue Mar 29 21:51:02 KST 2022
sdk.dir=C\:\\Users\\(username)\\AppData\\Local\\Android\\Sdk
ndk.dir=C\:\\Users\\(username)\\AppData\\Local\\Android\\Sdk\\ndk\\19.2.5345600

```

  - Use the following path for MacOS device.

```
sdk.dir=/Users/(username)/Library/Android/sdk
ndk.dir=/Users/(username)/Library/Android/sdk/ndk/19.2.5345600
```

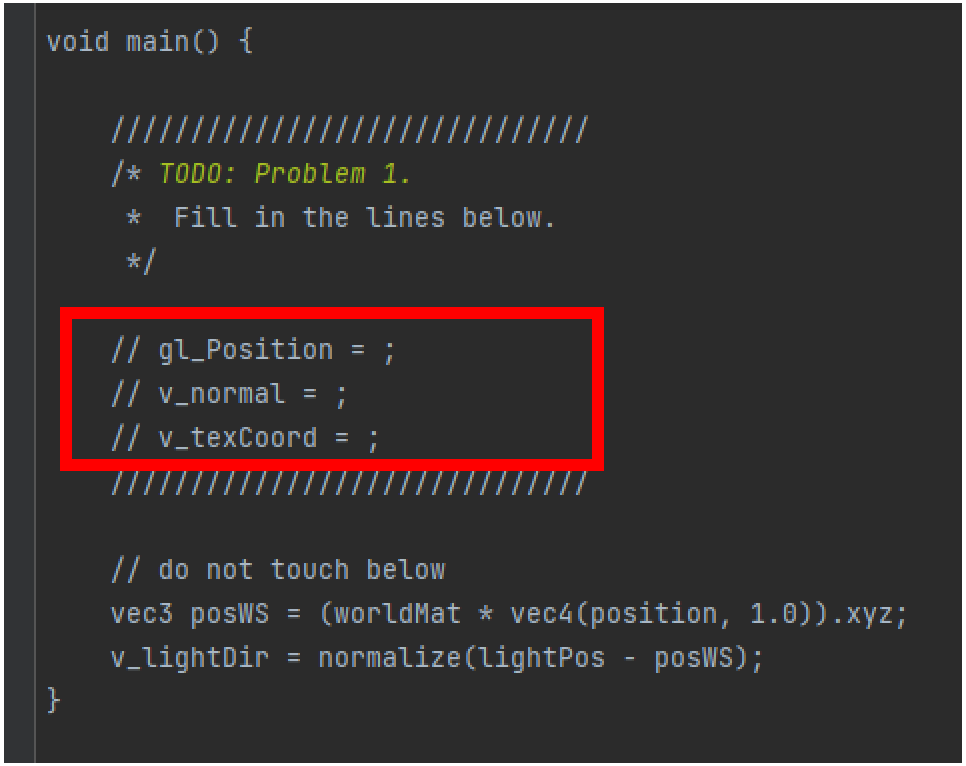
Gradle Sync

- After modifying, sync project with gradle files.
  - Files – Sync Project with Gradle Files



Problem 1

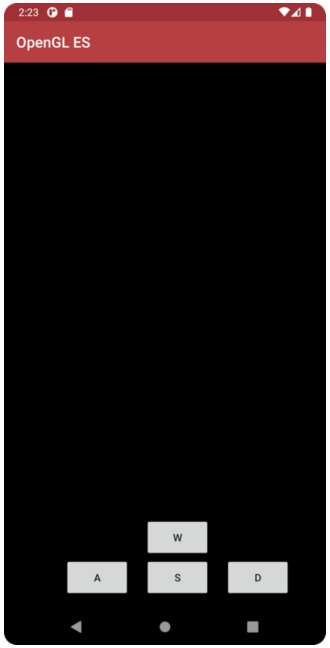
- Fill in some lines in the vertex shader.
    - `app/src/main/assets/vertex.glsl`



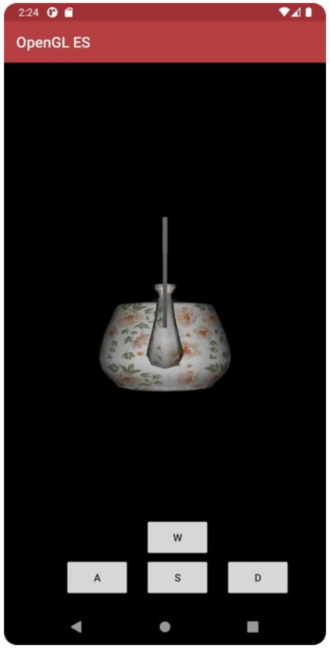
```glsl
void main() {

    //////////////////////////////
    /* TODO: Problem 1.
     *  Fill in the lines below.
     */

    // gl_Position = ;
    // v_normal = ;
    // v_texCoord = ;
    //////////////////////////////

    // do not touch below
    vec3 posWS = (worldMat * vec4(position, 1.0)).xyz;
    v_lightDir = normalize(lightPos - posWS);
}
```

Problem 1

- The clip-space **vertex position** must be calculated.
- The world-space **vertex normal** must be calculated accurately considering non-uniform scaling.
  - Note that the lighting will look weird if vertex normals are incorrect.
- The **texture coordinates** will be sent to the rasterizer stage with no change.



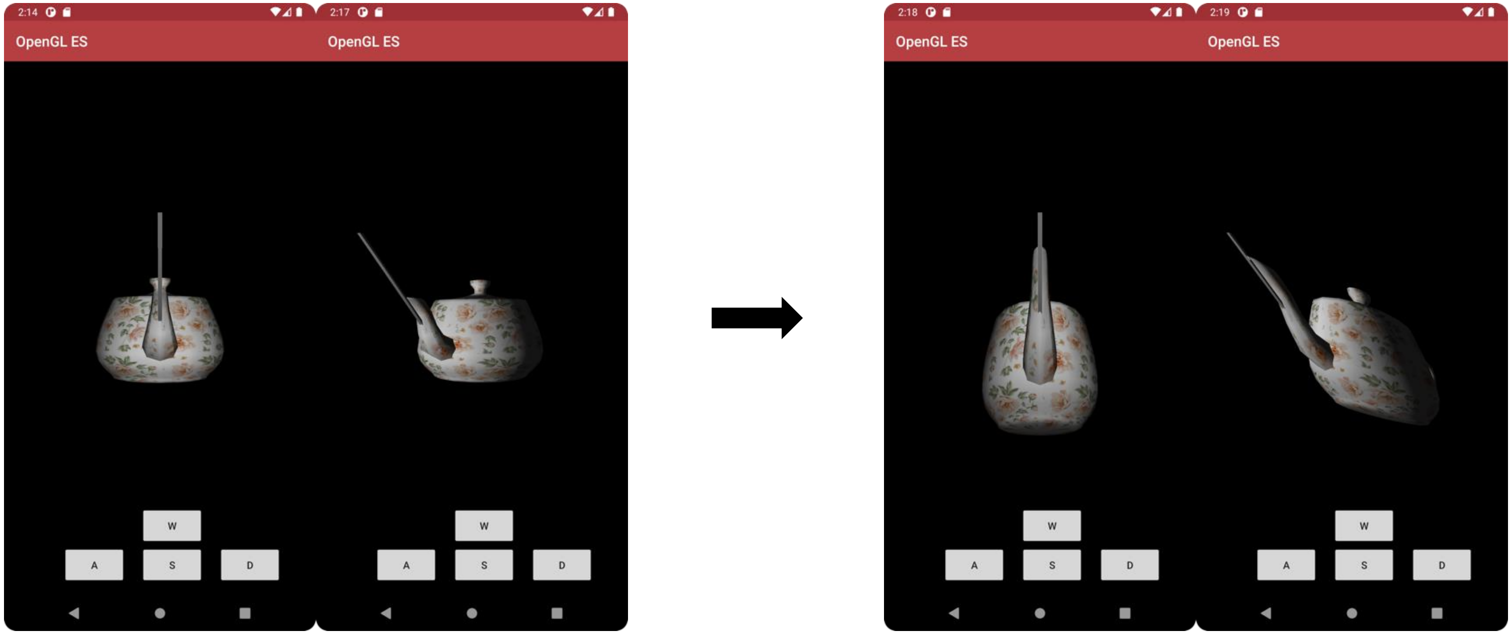
→



Problem 2

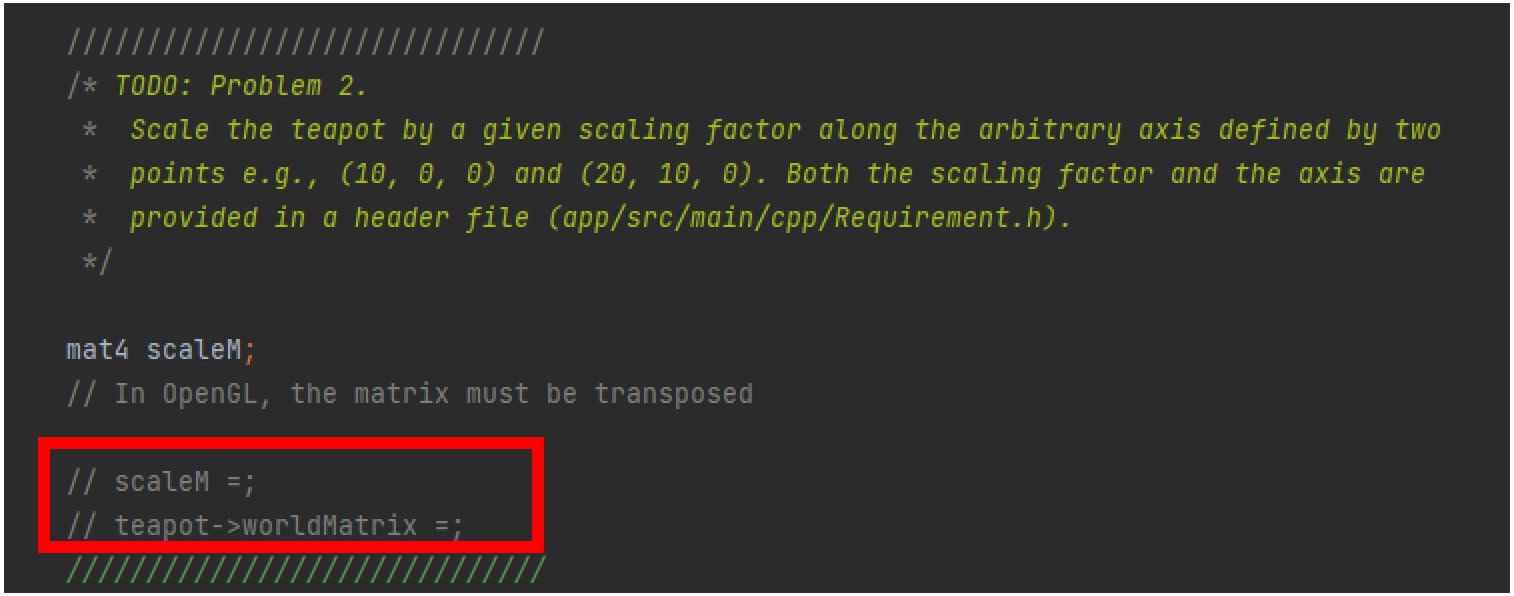
- Transform the teapot as follows:

  1. Scale the teapot by a given scaling factor along the arbitrary axis defined by two points, e.g., (10, 0, 0) and (20, 10, 0). Both the scaling factor and the axis are provided in a header file (app/src/main/cpp/Requirement.h).
     - Note that an angle can be represented in **degrees** or **radians**.
     - Hint: This scaling can be expressed as the **multiplication of five matrices**.



Problem 2

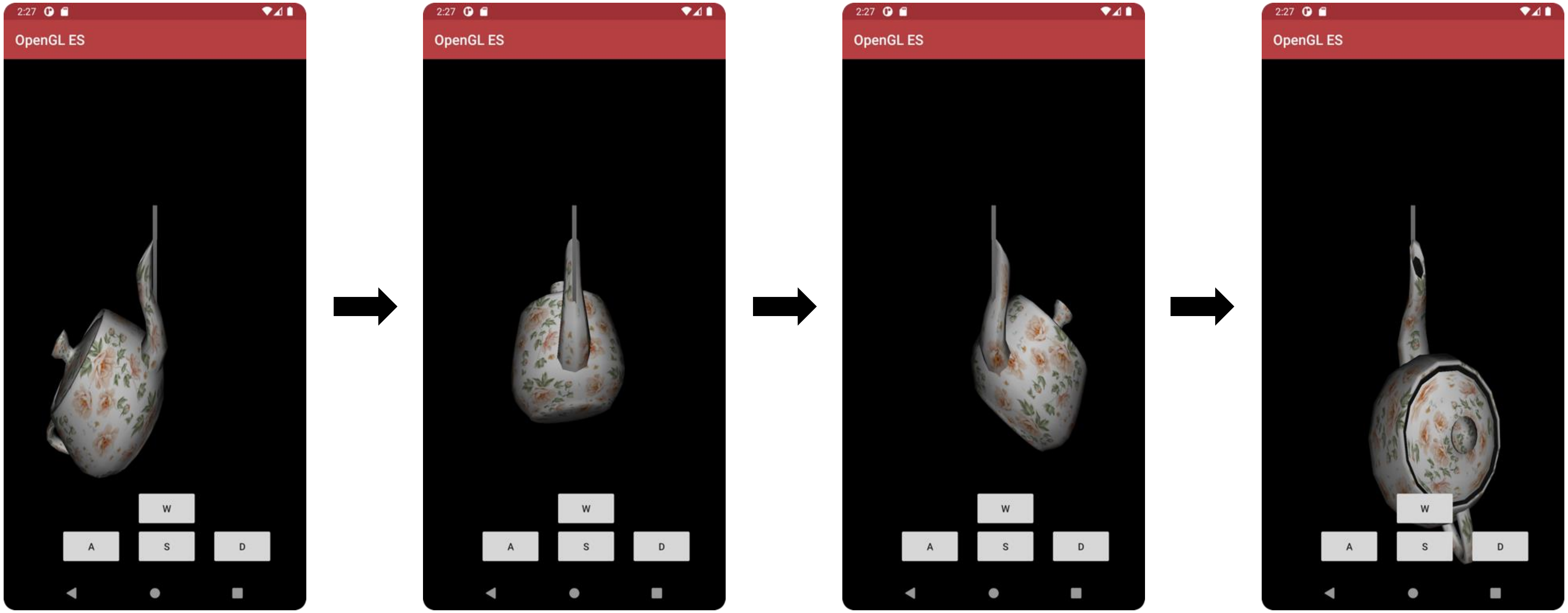
- Fill in some lines in `Scene::setup(…)`.
  - `app/src/main/cpp/scene.cpp`



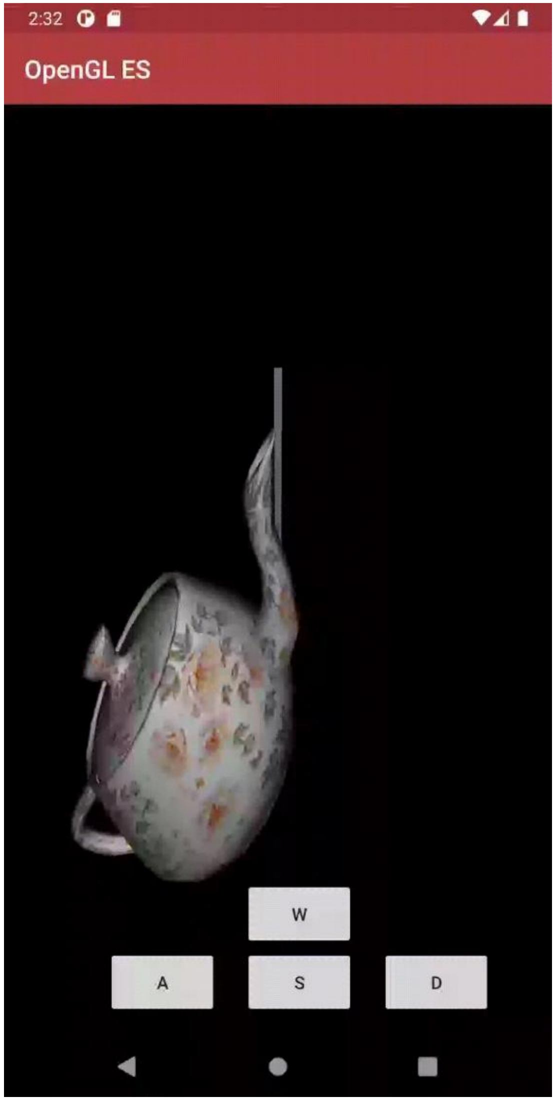
```
////////////////////////////////
/* TODO: Problem 2.
 *  Scale the teapot by a given scaling factor along the arbitrary axis defined by two
 *  points e.g., (10, 0, 0) and (20, 10, 0). Both the scaling factor and the axis are
 *  provided in a header file (app/src/main/cpp/Requirement.h).
 */

mat4 scaleM;
// In OpenGL, the matrix must be transposed

// scaleM =;
// teapot->worldMatrix =;
////////////////////////////////
```

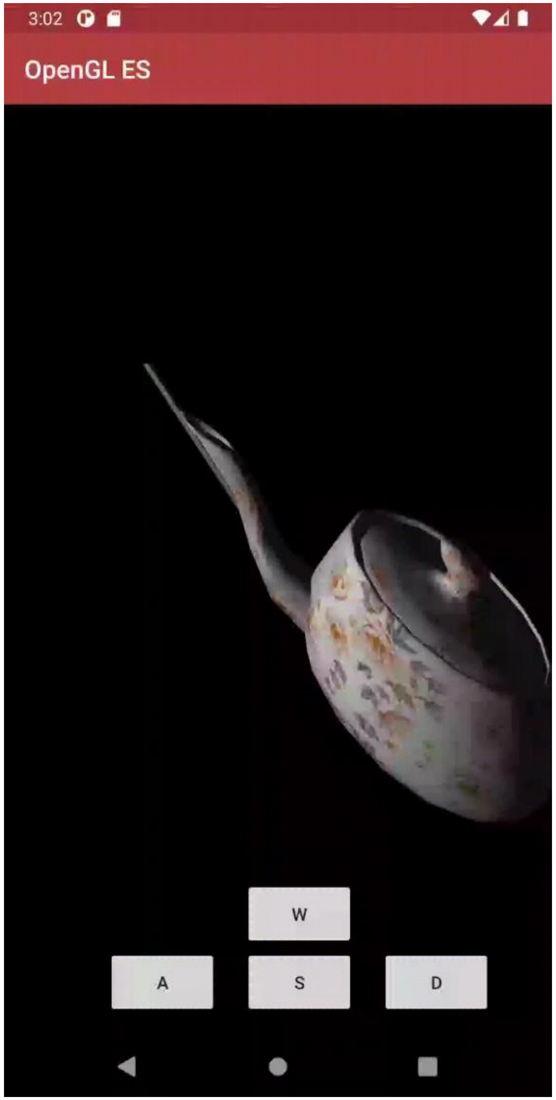
Problem 3

1. Keep rotating the teapot clockwise about the rotation axis defined by two points, e.g., (10, 0, 0) and (20, 10, 0). The axis is provided in a header file (app/src/main/cpp/Requirement.h).
   - Note that an angle can be represented in **degrees** or **radians**.
   - Hint: This rotation can be expressed as the **multiplication of five matrices**.



Problem 3

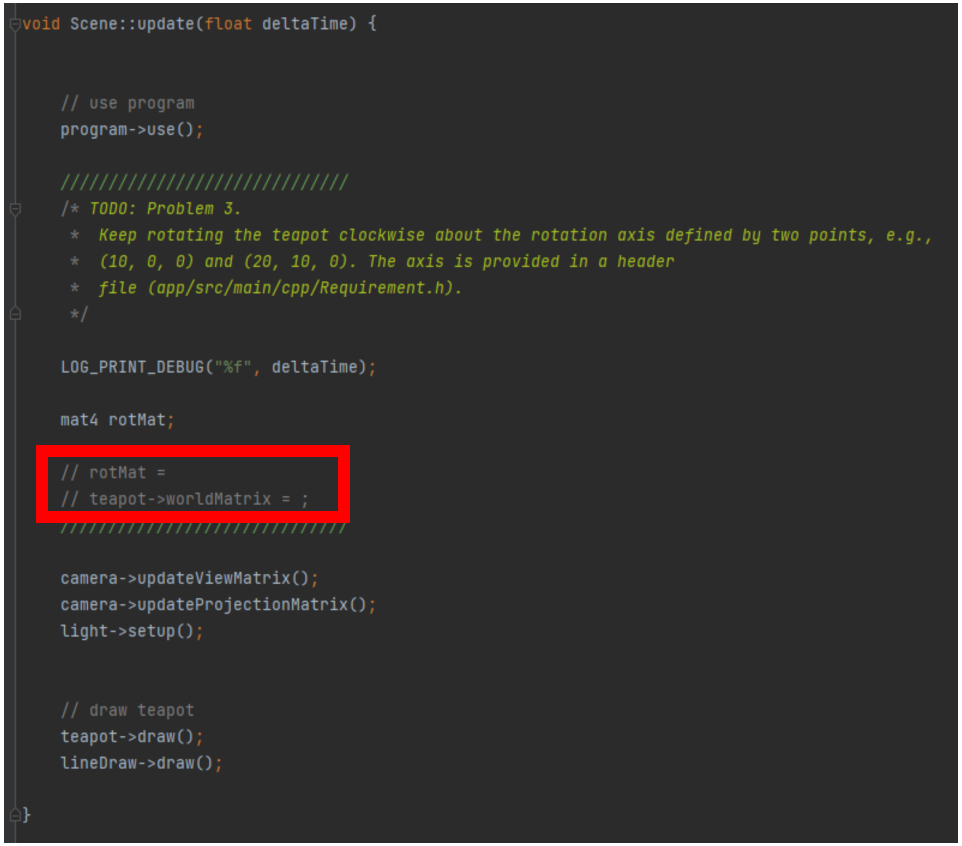




Problem 3

- Fill a line in `Scene::update(float deltaTime)`.
  - `app/src/main/cpp/scene.cpp`

    `deltaTime` is given in seconds



```cpp
void Scene::update(float deltaTime) {


    // use program
    program->use();

    //////////////////////////////
    /* TODO: Problem 3.
     *  Keep rotating the teapot clockwise about the rotation axis defined by two points, e.g.,
     *  (10, 0, 0) and (20, 10, 0). The axis is provided in a header
     *  file (app/src/main/cpp/Requirement.h).
     */

    LOG_PRINT_DEBUG("%f", deltaTime);

    mat4 rotMat;

    // rotMat =
    // teapot->worldMatrix = ;
    //////////////////////////////

    camera->updateViewMatrix();
    camera->updateProjectionMatrix();
    light->setup();


    // draw teapot
    teapot->draw();
    lineDraw->draw();

}
```

GLM Library

- You may want to use some glm functions.

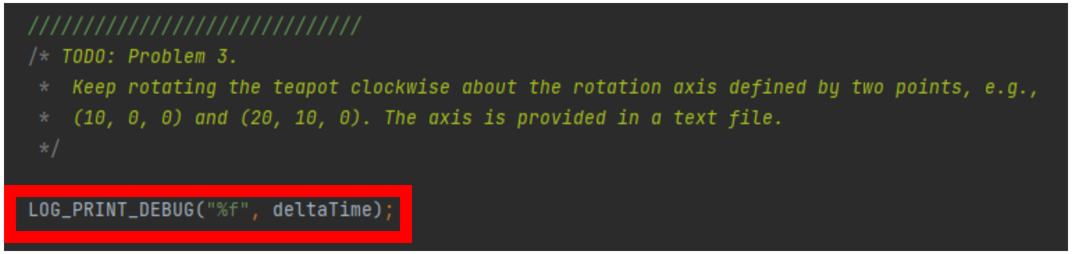
```
glm::cross(…)

glm::normalize(…)

glm::translate(…)

glm::rotate(…)

…
```

- You can use the buttons or drag the mouse to change the camera's perspective

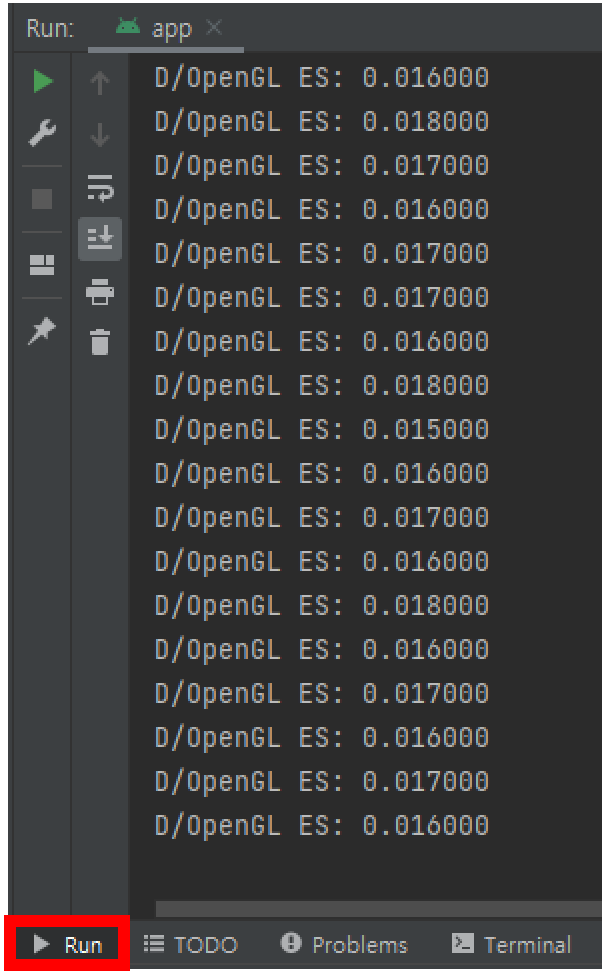
Debugging

- You can use `LOG_PRINT_DEBUG()` for debugging purposes

example)



```
///////////////////////////////
/* TODO: Problem 3.
 *  Keep rotating the teapot clockwise about the rotation axis defined by two points, e.g.,
 *  (10, 0, 0) and (20, 10, 0). The axis is provided in a text file.
 */

LOG_PRINT_DEBUG("%f", deltaTime);
```



```
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.018000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.018000
D/OpenGL ES: 0.015000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.018000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.016000
D/OpenGL ES: 0.017000
D/OpenGL ES: 0.016000
```

Submission

- Deadline
  - April 18 (Wed) 19:00
- Submission files ({student_id}_{name}.zip)
  - Vertex shader file (app/src/main/assets/vertex.glsl)
  - Scene class file (app/src/main/cpp/scene.cpp)
- Submission to Blackboard
- Contact
  - TA email: 2024.cg.ta@gmail.com