Computer Graphics, Lab Assignment 3

Handed out: April 1, 2020

Due: 23:59, April 1, 2020 (NO SCORE for late submissions!)

- *Only accept answers submitted via git push to this course project for you at https://hconnect.hanyang.ac.kr (<Year>_<Course no.>_<Class code>/<Year>_<Course no.>_<Student ID>.git).*
- *Place your files under the directory structure <Assignment name>/<Problem no.>/<your file> just like the following example.*

```
+ 2020_ITE0000_2019000001
  + LabAssignment2/
    + 1/
      - 1.py
    + 2/
      - 2.py
    + 3/
      - 3.py
```

- *The submission time is determined not when the commit is made but when the git push is made.*

1. Write down a Python program to draw a rotating triangle.
   A. Set the window title to **your student ID** and the window size to (480,480).
   B. Draw a triangle using render() function below (DO NOT modify it!).

```
def render(T):
    glClear(GL_COLOR_BUFFER_BIT)
    glLoadIdentity()
    # draw cooridnate
    glBegin(GL_LINES)
    glColor3ub(255, 0, 0)
    glVertex2fv(np.array([0.,0.]))
    glVertex2fv(np.array([1.,0.]))
    glColor3ub(0, 255, 0)
    glVertex2fv(np.array([0.,0.]))
    glVertex2fv(np.array([0.,1.]))
    glEnd()
    # draw triangle
    glBegin(GL_TRIANGLES)
    glColor3ub(255, 255, 255)
    glVertex2fv( (T @ np.array([.0,.5,1.]))[:-1] )
    glVertex2fv( (T @ np.array([.0,.0,1.]))[:-1] )
    glVertex2fv( (T @ np.array([.5,.0,1.]))[:-1] )
    glEnd()
```

C. Expected result: Uploaded LabAssignment3-1.mp4

   i. Do not mind the initial angle of the triangle.

D. The triangle should be t rad rotated when t seconds have elapsed since the program was executed.

E. You need to somehow combine a rotation matrix and a translation matrix to produce the expected result.

F. Files to submit: A Python source file (Name the file whatever you want (in English). Extension should be .py)

2. Write down a Python program to draw a transformed triangle.

A. Set the window title to **your student ID** and the window size to (480,480).

B. Draw a triangle using render() function of prob 1 (DO NOT modify it!).

C. If you press or repeat a key, the triangle should be transformed as shown in the Table:

Key	Transformation
W	Scale by 0.9 times in x direction
E	Scale by 1.1 times in x direction
S	Rotate by 10 degrees counterclockwise
D	Rotate by 10 degrees clockwise
X	Shear by a factor of -0.1 in x direction

C	Shear by a factor of 0.1 in x direction
R	Reflection across x axis
1	Reset the triangle with identity matrix

D. Transformations should be accumulated (composed with previous one) unless you press '1'.

   i. Be sure: gComposedM = newM @ gComposedM

   ii. You'll need to make 'gComposedM' as a global variable.

E. Files to submit: A Python source file (Name the file whatever you want (in English). Extension should be .py)

F. Expected result: