**COMP 3069 -- Computer Graphics**

**Assignment 1**

(Due time: Noon, Tuesday, 6 November 2018)

**PART I: Written part (30%)**

1) There is a cube drawn at the origin of the coordinate systemoXYZ. We want to rotate the cube by 30 degrees clockwise around a vector U, which originates from P1(2,3,0) on the XoY plane and passes through point P2(3,4,1) in the space.

a) Suppose that through the above transformation, the homogeneous coordinates of any point p of the given cube are transformed to the target point p', and p' = M p where M is a 4x4 matrix. Specify what sub-matrices is M composed of, and compute the values of the entries of the 4x4 matrix M. **(20 credits)**

b) Write OpenGL code to represent the above transformation. Here we suppose that the cube is drawn at origin using the procedure DrawCube(). **(10 credits)**

**PART II: Programming Part (Hierarchical Modelling) (70%)**

1. Create a static mannequin using spheres and cylinders using hierarchical modeling approach. The structure should be like the figure 1 and figure 2. There are few requirements:
   1. Leg is thicker than arm
   2. Different parts have different color (e.g. arm=red, leg=green, joint=blue, etc.)

**(30 credits)**

Figure 1: The hierarchical structure of mannequin.

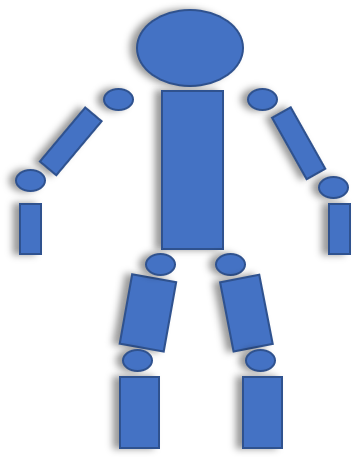


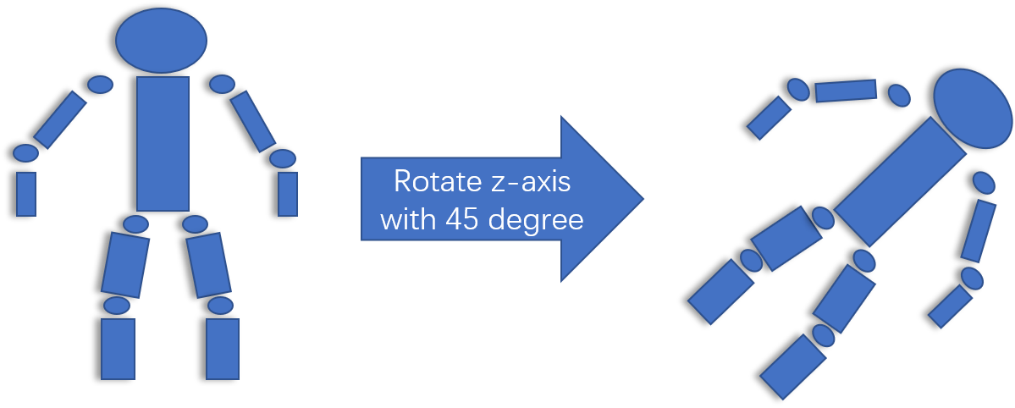
Figure 2: Simple example.

1. Dynamic limb control:

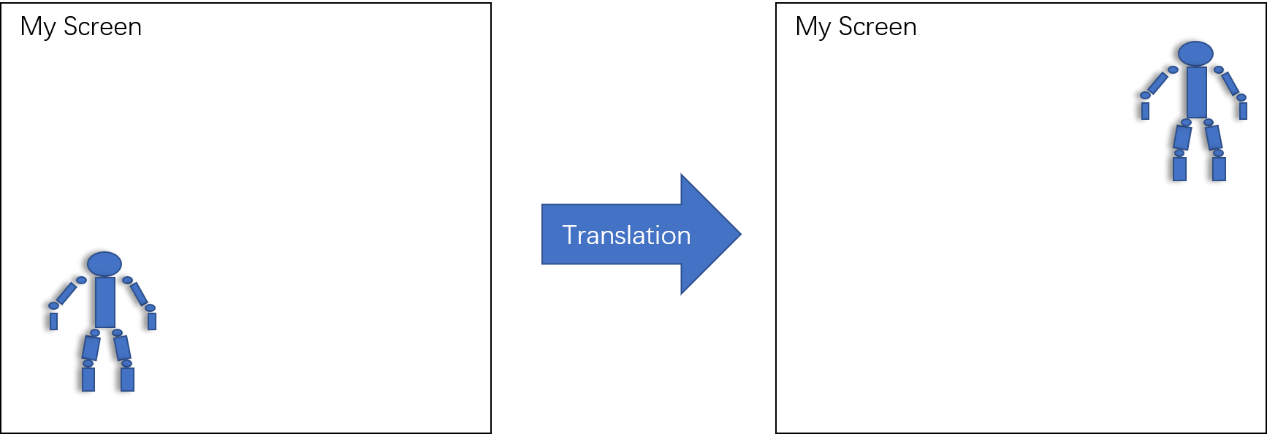
In this section you need to build a dynamic control function. You need to implement the interactive functions using the keyboard. There are several aims to achieve:

* Rotate control of the whole mannequin.

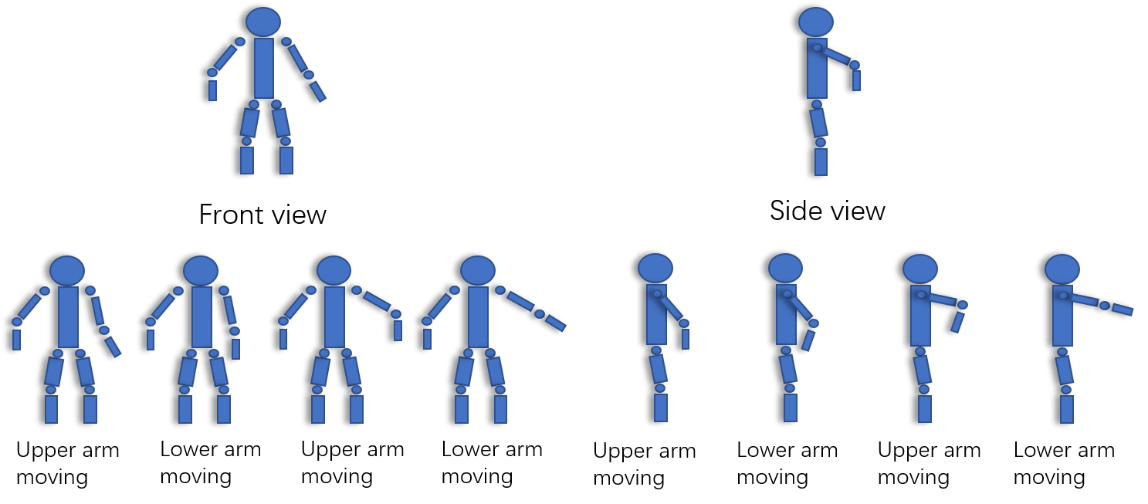
The mannequin can rotate around the x-axis, y-axis and z-axis. The control interface can be: (1) “R+X” to rotate the mannequin around x-axis, (2) “R+Y” to rotate the mannequin around y-axis, (3) “R+Z” to rotate the mannequin round z-axis.



* Translation (or shift) control of the whole mannequin (x, y, z)



* Moving control of left arm including its lower arm and upper arm. In this part, you only need to implement the function to left arm. You need to implement the interactive functions using the keyboard to control the action of left arm. For example, if the left upper arm is going up, the left lower arm will go up with the upper arm. In contrast, if the lower arm is going up, the upper arm does not necessarily go up with lower arm. The action definitions are shown below:



* Reset the space and mannequin

There should be a button to set the working space back to the default state.

**(20 credits)**

1. Left elbow control:

There are limitations of human joints, which have rotation ranges. For convenience of your programming, your programming only need to concentrate on the lefe elbow joint. Please limit the rotation range of left elbow to be 5-180 degree in your grogramming. So you only need to apply the rotation range to the left elbow of the left arm.

**(10 credits)**

1. Report:

A report needs to be submitted, and it is required to include:

* 1. Describe the working environment of your code (OpenGL version, GLUT version, Freeglut version, Windows version, MacOS version, Visual Studio version, Xcode version, Cmake version, other packaged you used)
  2. Describe how you design the hierarchical structure of mannequin.
  3. The structure of your code (e.g. flowchart)
  4. The operation instruction (e.g. how to move the limbs)
  5. The screenshot of each section’s output.

**(10 credits)**

**Submission instructions:**

1. Submit the code with report
2. Your program and source code need to be submitted. Due to the upload limitation of Moodle, you need to upload the following files:
   1. “.cpp”, “.h”, and other related files
   2. an executable .exe file (using release function)
3. If your code is not executable, you may be required an in-class demonstration.

**Plagiarism warning**

The report and code will be uploaded and checked by Turnitin, do not copy the code from any open source websites.