Tut 3

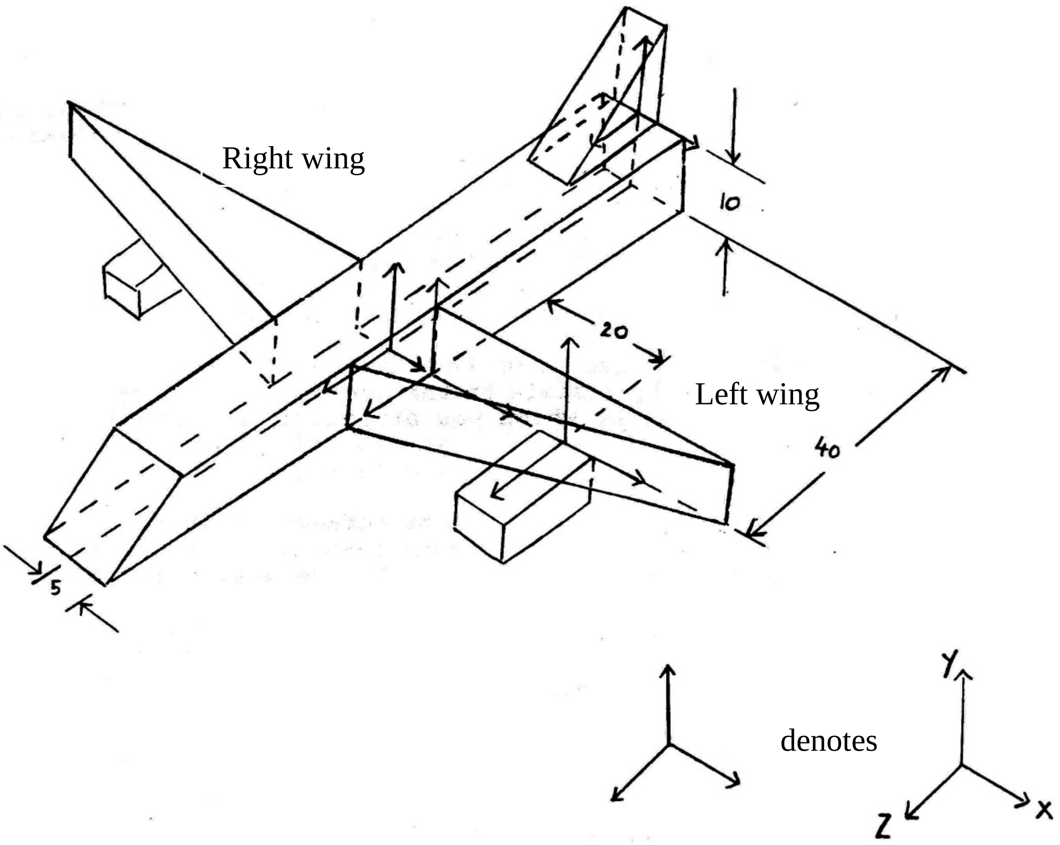
Qn 1

A geometric model AIRPLANE consists of a plane body, a right wing, a left wing and a rear wing. Under the right and the left wings are the right and the left engines respectively.

Assume that the functions *body()*, *wing()*, *rear_wing()* and *engine()* have been written. Calling these functions will generate the respective parts, defined in their own coordinate systems. *wing( )* will draw the left wing.

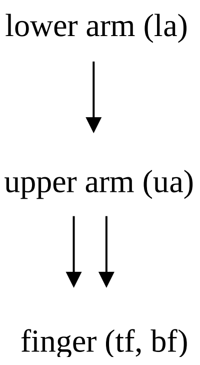
a) Design the DAG.

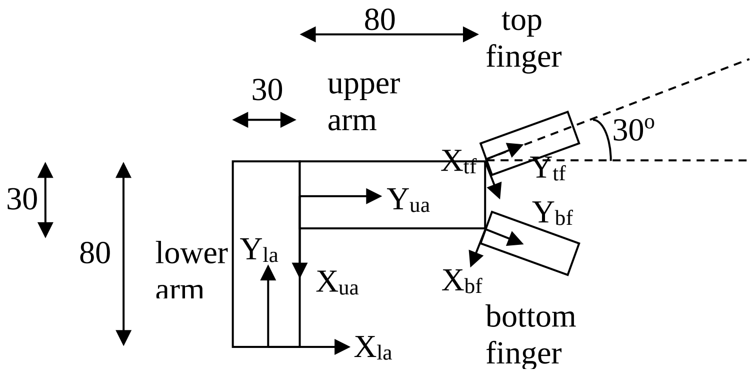
b) Write an OpenGl program for the model.



Qn 2

A robotic hand, along with its directed acyclic graph (DAG), is shown below:





The coordinate systems of the lower arm and the upper arm involve no scaling.

The coordinate systems of the (top and bottom) fingers are scaled down by 0.5 compared with the upper arm coordinate system.

The arms and fingers are rectangular boxes. The length (i.e. X dimension) and width (i.e. Y dimension) are as shown. The height (i.e. Z dimension) of the arm boxes is 50. The height of the finger boxes is 25. Note that for all the parts, the Z axis is pointing out of the paper.

a) Starting from the *glutSolidCube* function, implement a *box (length, width, height)* function that can be used to draw the arms and fingers.

   Note that the coordinate system of the box involves a translation in Y from the original coordinate system.

b) Find the coordinate transformations

   i) $\mathbf{M}_{la \leftarrow ua}$ (i.e., from upper arm to lower arm),
   ii) $\mathbf{M}_{ua \leftarrow tf}$ (i.e., from top finger to upper arm),
   iii) $\mathbf{M}_{ua \leftarrow bf}$ (i.e., from bottom finger to upper arm)

   and hence write the OpenGL program for the robotic hand.

c) Suppose the robotic hand moves such that

   - the lower arm is rotated about its Z axis by 30°
   - the top finger is rotated about its Z axis by 20°

   Add code into your program to do that. Underline your newly added code.

OpenGL Mini-project Progress

At this point, you should be programming OpenGL Ex 2.

You should also be planning your main character in your animation, which should be a hierarchy with more than three parts. Create an original character.

Define its DAG and the range of its motion. Then do the programming. Note that no matter how complicated is your objects and your scene, everything should be in the function with the name you define in

glutDisplayFunc (your_function_name);