**Assignment 1**

**Purpose:**

* Practice in
  + 3-dimensional composite transforms.
  + 2-dimensional window to viewport mapping

**Task:**

Add a "Load" command (button) to Assignment\_00 such that the program will be able to read the data from an input file and display the data on the viewport.

Details:

* **Load (Open) Input File**

This command loads all the data and parameters from a text file. The system should prompt for the name of the input  file. Each  line of the file will contain either a vertex, a face, or a viewing parameter definition as described below:

v <x1> <y1> <z1>  //Define a vertex

v <x2> <y2> <z2>

.

.

.

v <xn> <yn> <zn>

f <k1> <l1> <m1> //Define a face (k, l , and m are integers corresponding to the vertex number)

f <k2> <l2> <m2>

.

.

.

f <km> <lm> <mm>

w <xmin> <ymin> <xmax> <ymax>   // Define 2D Window

s <xmin><ymin><xmax> <ymax. //Define viewport (normalized coordinates)

Notes:

* Each "v" line defines a new vertex with the given x,y,z coordinates.
* Each vertex is given a unique identifier starting from 1 (*not 0*).
* Each "f" line defines a new triangular face whose corners are the vertices with the given identifiers u,v,w.
* An s line defines the viewport in the normalized coordinates
* No clipping is required for this lab.

For simplicity this lab assumes a parallel projection with direction of projection being parallel to the z axis with no clipping in  the 3-  dimensional world (these features will be added in the future labs).

It is assumed that all data are given in right-handed  coordinate  system. The positive direction of rotation is counter clock-wise when viewed from plus infinity.

Your program should display the data by simply dropping (ignoring) the z coordinates of each vertex (map your data to 2- dimensional world coordinate system), and then map data from 2D window to viewport. You do not need to implement  clipping for this lab.

Your program should draw the boundaries of the viewport.