**Assignment 3**

Purpose:

* Practice in three-dimensional transforms.
* Practice in three-dimensional clipping.
* Parallel projection mapping.

This assignment extends assignments 01 and 02  to include 3d clipping and parallel projections.

Write a graphic package to include the following commands.

* **Browse**  
  This command should allow the user to select an input file .
* **Load (Open) Input File**  
  This command loads all the data and parameters from the input text file and displays the objects using parallel projection. All objects should be clipped in 3-d and displayed in the viewport. Each line of the file will contain  either a vertex, a  face, or a  viewing parameter definition as described below:
* **Rotate around x axis** **by D degrees in N incremental steps (Same as previous assignment)**  
  This command rotates all the objects around line x axis by D degrees in N incremental steps. The intermediate results  should be  displayed
* **Rotate around y axis by D degrees in N incremental steps (Same as previous assignment)**  
  This command rotates all the objects around line y axis by D degrees in N incremental steps. The intermediate results  should be  displayed
* **Rotate around z axis  by D degrees in N incremental steps (Same as previous assignment)**  
  This command rotates all the objects around line z axis by D degrees in N incremental steps. The intermediate results  should be  displayed
* **Scale Sx, Sy, Sz around point Ax,Ay,Az in  N incremental steps (Same as previous assignment)**  
  This command scales all the objectsaround point A(x,y,z)  by Sx, Sy, and Sz in N incremental steps. The intermediate results should be  displayed
* **Translate Tx, Ty, Tz, N**

This command translates all the objects by Tx, Ty, and Tz in N incremental steps. The intermediate results should be displayed

* **Fly XA, YA, ZA, XB, YB, ZB, N**

This command moves VRP from Point A to point B in N incremental steps. VPN and VUP should stay the same. The intermediate results should be displayed

Detail format 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 <u1> <v1> <w1> //Define a face (u,v, and w are integers corresponding to the vertex number))

f <u2> <v2> <w2>

.

.

.

f <um> <vm> <wm>

r <x> <y> <z>   //Define VRP (WC)

n <x> <y> <z>  //Define VPN (WC)

u <x> <y> <z>  //Define VUP (WC)

p <x> <y> <z>  //Define PRP (VRC)

w <umin><umax><vmin><vnax><nmin><nmax>    //Define View Volume (VRC)

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
    - The set of the r, n, u, p, w, and s define the viewing parameters and viewport for the parallel projection.
    - Your program must include 3-dimensional clipping.
    - Boundaries of the viewports must be displayed.
    - The user should be able to resize the canvas and the content of the canvas should be automatically refreshed proportionally.