**Practical 8**

Implement animation principles for any object.

**Program Code:-**

#include <GL/gl.h>

#include <GL/glut.h>

#include <math.h>

//global variable diclaration

int frameNumber = 0; //frame no

void drawWindmill() //Function to draw windmill

{

int i;

glColor3f(1.0,1.0,0.0); //red green blue

glBegin(GL\_POLYGON);

glVertex2f(-0.05f, 0); //for drawing rectangular base part

glVertex2f(-0.05f, 3);

glVertex2f(0.05f, 3);

glVertex2f(0.05f, 0);

glEnd();

glTranslatef(0,3,0); //x,y,z

glColor3f(1.0,0.0,0.0); //red,green,blue (RED PLATES OF WINDMILL)

glRotated(frameNumber \* (180.0/45), 0, 0, 1); //(angle,x,y,z)

for (i = 0; i < 4; i++) //LOOP TO DRAW FOUR PLATES

{

glRotated(90, 0, 0, 1); //90,0,0,Z

glBegin(GL\_POLYGON);

glVertex2f(0,0); //FOR DRAWING TYIANGLULAR PLATE

glVertex2f(1.0f, 0.2f);

glVertex2f(1.0f,-0.2f);

glEnd();

}

}

void display() //DISPLAY FUNCTION

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glLoadIdentity(); //TAKES IDENTITY MATRIX

glPushMatrix(); //PUSH MATRIX

glTranslated(2.2,1.6,0); //SET POSITION OF WINDMILL

glScaled(0.4,0.4,1); //SCALLING WINDMILL WITH POINT (0.4,0.4,1)

drawWindmill(); //FUNCTION CALL TO DRAW WINDMILL

glPopMatrix(); //POP MATRIX

glPushMatrix(); //PUSH MATRIX

glTranslated(3.7,0.8,0); //SET POSITION OF WINDMILL

glScaled(0.7,0.7,1); //SCALLING WINDMILL WITH POINT(0.7,0.7,1)

drawWindmill(); //FUNCTION CALL TO DRAW WINDMILL

glPopMatrix(); //POP MATRIX

glutSwapBuffers(); //SWAP BUFFER

}

void doFrame(int v)

{

frameNumber++; //INCREMENT FRAME NO

glutPostRedisplay(); //POST REDISPLAY

glutTimerFunc(10,doFrame,0);

}

void init() //FUNCTION INITIALISATION

{

glClearColor(0,0,0,0);

glMatrixMode(GL\_PROJECTION); //MATRIX MODE FOR PROJECTION

glLoadIdentity(); //LOADS IDENTITY MATRIX

glOrtho(0, 7, -1, 4, -1, 1); //MIN X,MAX X,MIN Y,MAX Y,MIN Z,MAX Z VALUE

glMatrixMode(GL\_MODELVIEW); //MATRIX MODE FOR MODEL VIEW

}

int main(int argc, char\*\* argv) //MAIN FUNCTION

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DOUBLE);

glutInitWindowSize(700,500); //DEFINED WINDOW SIZE 700\*500

glutInitWindowPosition(100,100); //DEFINED WINDOW POSITION 100,100

glutCreateWindow("WINDMILL"); //NAME OF WINDOW

init(); //FIRSTLY CALL TO INTIALISE VALUE

glutDisplayFunc(display); //DISPLAY

glutTimerFunc(200,doFrame,0); //TIMER FUNC

glutMainLoop();

return 0;

}