Name : Niranjan Vinod Patil.

**Title :** Develop a program to draw 2D cube and perform the transformation on it using openGL.

Code -

#include<GL/glut.h>

#include<GL/gl.h>

#include<stdio.h>

float x1,x2,y1,y2,deg;//rec variables

int ch ; //choice variable

float xx,yy ; // translation varaibles

float px,py ; // scalling varaibles

void dda(float x1, float y1,float x2, float y2)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_POINTS);

glColor3f(0.0f,1.0f,0.0f);

float dx, dy, len,i,xinc,yinc;

dx=abs(x2-x1);

dy=abs(y2-y1);

if(dx>=dy)

len = dx;

else

len = dy;

xinc=(x2-x1)/len;

yinc=(y2-y1)/len;

for(i=0; i<=len; i++)

{

x1=x1+xinc;

y1=y1+yinc;

glVertex2f(x1,y1);

}

}

void scaling(void){

//draw rectangle here

dda(x1,y1,x2,y1);

dda(x2,y1,x2,y2);

dda(x2,y2,x1,y2);

dda(x1,y2,x1,y1);

dda(x1\*px,y1\*py,x2\*px,y1\*py);

dda(x2\*px,y1\*py,x2\*px,y2\*py);

dda(x2\*px,y2\*py,x1\*px,y2\*py);

dda(x1\*px,y2\*py,x1\*px,y1\*py);

glEnd();

glFlush();

}

void rotation(void){

glClear(GL\_COLOR\_BUFFER\_BIT);

glPushMatrix();

glBegin(GL\_POLYGON);

glVertex3f (x1,y1, 0.0);

glVertex3f (x2, y1, 0.0);

glVertex3f (x2, y2, 0.0);

glVertex3f (x1, y2, 0.0);

glEnd();

glRotatef(deg,0.0,0.0,1.0);

glBegin(GL\_POLYGON);

glVertex3f (x1,y1, 0.0);

glVertex3f (x2, y1, 0.0);

glVertex3f (x2, y2, 0.0);

glVertex3f (x1, y2, 0.0);

glEnd();

glPopMatrix();

//glFlush();

glutSwapBuffers();

}

void translation(void)

{

//draw rectangle here

dda(x1,y1,x2,y1);

dda(x2,y1,x2,y2);

dda(x2,y2,x1,y2);

dda(x1,y2,x1,y1);

dda(x1+xx,y1+yy,x2+xx,y1+yy);

dda(x2+xx,y1+yy,x2+xx,y2+yy);

dda(x2+xx,y2+yy,x1+xx,y2+yy);

dda(x1+xx,y2+yy,x1+xx,y1+yy);

glEnd();

glFlush();

}

void Init()

{

glClearColor(1.0,1.0,1.0,0);

glColor3f(0.0,0.0,0.0);

gluOrtho2D(0, 500, 0, 500);

}

int main(int argc, char \*argv[]){

printf("Enter the value of x1,y1\n");

scanf("%f",&x1);

scanf("%f",&y1);

printf("Enter the value of x2,y2\n");

scanf("%f",&x2);

scanf("%f",&y2);

printf("\n1. Translation \n2. Scaling \n3. Rotation\n\nEnter your choice :\n");

scanf("%d",&ch);

glutInit(&argc, argv);

glutInitWindowSize(500,500);

glutInitWindowPosition(100,100);

switch (ch) {

case 1 :{

printf("Enter the value of next points to where object will be moved");

scanf("%f",&xx);

scanf("%f",&yy);

glutCreateWindow("2D Transformation ( translation )");

Init();

glutDisplayFunc(translation);

glutMainLoop();

break ;}

case 2 :{

printf("Enter the value of Scalling factor (x,y)");

scanf("%f",&px);

scanf("%f",&py);

glutCreateWindow("2D Transformation ( scaling )");

Init();

glutDisplayFunc(scaling );

glutMainLoop();

break ;}

case 3 :{

printf("Enter the value of angle(in degrees) for roation :");

scanf("%f",&deg);

glutCreateWindow("2D Transformation ( rotation )");

Init();

glutDisplayFunc(rotation);

glutMainLoop();

break ;}

}

return 0;

}

Output:

Enter the value of x1,y1

50

100

Enter the value of x2,y2

100

150

1. Translation

2. Scaling

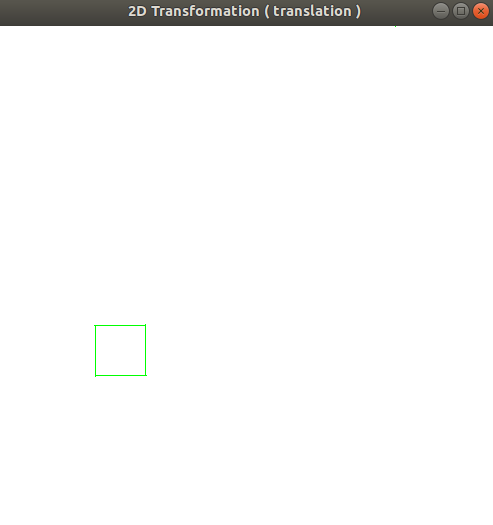
3. Rotation

Enter your choice :

2

Enter the value of Scalling factor (x,y)100

200



Enter the value of x1,y1

200

200

Enter the value of x2,y2

300

300

1. Translation

2. Scaling

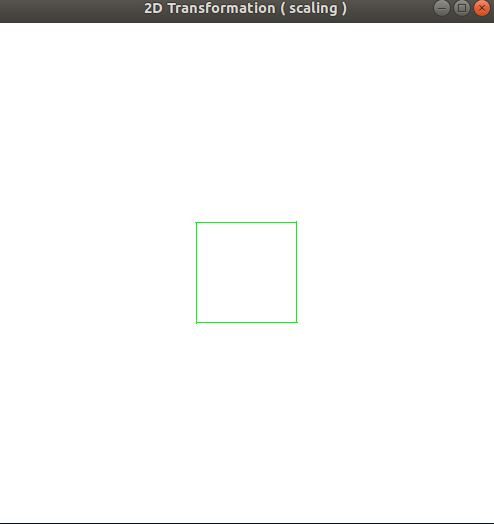
3. Rotation

Enter your choice :

2

Enter the value of Scalling factor (x,y)700

800



Enter the value of x1,y1

100

150

Enter the value of x2,y2

150

200

1. Translation

2. Scaling

3. Rotation

Enter your choice :

3

Enter the value of angle(in degrees) for roation :65

