**/\* Program No. :**

**Aim : WAP for rotation of a 3D object about an arbitrary axis.**

**\*/**

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<math.h>

#include<stdlib.h>

int xp[2],yp[2],z;

void display()

{

int x3,y3,x4,y4;

line(getmaxx()/2,0,getmaxx()/2,getmaxy());

line(0,getmaxy()/2,getmaxx(),getmaxy()/2);

settextstyle(SMALL\_FONT,HORIZ\_DIR,4);

outtextxy(getmaxx()/2+5,getmaxy()/2+5,"(0,0)");

outtextxy(getmaxx()-40,getmaxy()/2+5,"X Axis");

outtextxy(getmaxx()/2+10,5,"Y Axis");

outtextxy(5,getmaxy()/2+5,"X' Axis");

outtextxy(getmaxx()/2+10,getmaxy()-15,"Y' Axis");

rectangle(getmaxx()/2+xp[0],getmaxy()/2-yp[0],getmaxx()/2+xp[1],getmaxy()/2-yp[1]);

if(z>=xp[0])

{

x3=z+xp[0];

y3=z+yp[0];

x4=z+xp[1];

y4=z+yp[1];

rectangle(getmaxx()/2+x3,getmaxy()/2-y3,getmaxx()/2+x4,getmaxy()/2-y4);

line(getmaxx()/2+xp[0],getmaxy()/2-yp[0],getmaxx()/2+x3,getmaxy()/2-y3);

line(getmaxx()/2+xp[1],getmaxy()/2-yp[1],getmaxx()/2+x4,getmaxy()/2-y4);

line(getmaxx()/2+xp[0],getmaxy()/2-yp[1],getmaxx()/2+x3,getmaxy()/2-y4);

line(getmaxx()/2+xp[1],getmaxy()/2-yp[0],getmaxx()/2+x4,getmaxy()/2-y3);

}

else

{

x3=xp[0]-z;

y3=yp[0]-z;

x4=xp[1]-z;

y4=yp[1]-z;

rectangle(getmaxx()/2+x3,getmaxy()/2-y3,getmaxx()/2+x4,getmaxy()/2-y4);

line(getmaxx()/2+xp[0],getmaxy()/2-yp[0],getmaxx()/2+x3,getmaxy()/2-y3);

line(getmaxx()/2+xp[1],getmaxy()/2-yp[1],getmaxx()/2+x4,getmaxy()/2-y4);

line(getmaxx()/2+xp[0],getmaxy()/2-yp[1],getmaxx()/2+x3,getmaxy()/2-y4);

line(getmaxx()/2+xp[1],getmaxy()/2-yp[0],getmaxx()/2+x4,getmaxy()/2-y3);

}

}

void rotation()

{

float res[4][1],p[4][4],t[4][1],rad;

int ang,i,j,k,l,rch;

for(i=0;i<4;i++)

for(j=0;j<4;j++)

p[i][j]=(i==j);

printf("\nEnter the rotating angle : ");

scanf("%d",&ang);

rad=ang\*0.0174;

printf("\nChoose the axis of roration ");

printf("\n\n1.X-axis");

printf("\n2.Y-axis");

printf("\n3.Z-axis");

printf("\n\nEnter your choice (1-3) : ");

scanf("%d",&rch);

switch(rch)

{

case 1:p[1][1]=cos(rad);

p[1][2]=(-1)\*sin(rad);

p[2][1]=sin(rad);

p[2][2]=cos(rad);

break;

case 2:p[0][0]=cos(rad);

p[2][0]=(-1)\*sin(rad);

p[0][2]=sin(rad);

p[2][2]=cos(rad);

break;

case 3:p[0][0]=cos(rad);

p[0][1]=(-1)\*sin(rad);

p[1][0]=sin(rad);

p[1][1]=cos(rad);

break;

default:printf("\nInvalid Choice!");

getch();

exit(1);

}

clrscr();

cleardevice();

for(i=0;i<2;i++)

{

t[0][0]=xp[i];

t[1][0]=yp[i];

t[2][0]=z;

t[3][0]=1;

for(j=0;j<4;j++)

{

for(k=0;k<1;k++)

{

res[j][k]=0;

for(l=0;l<4;l++)

res[j][k]=res[j][k]+(p[j][l]\*t[l][k]);

}

}

xp[i]=res[0][0];

yp[i]=res[1][0];

z=res[2][0];

}

display();

}

void main()

{

int ch,i;

int gdriver = DETECT, gmode, errorcode;

initgraph(&gdriver, &gmode, "C:\\TC3.0\\BGI");

errorcode = graphresult();

clrscr();

if (errorcode != grOk)

{

printf("Graphics error: %s\n", grapherrormsg(errorcode));

printf("Press any key to exit.");

getch();

exit(1);

}

printf("\nEnter co-ordinate of starting vertex, x :");

scanf("%d",&xp[0]);

printf("\nEnter co-ordinate of starting vertex, y :");

scanf("%d",&yp[0]);

printf("\nEnter co-ordinate of ending vertex, x :");

scanf("%d",&xp[1]);

printf("\nEnter co-ordinate of ending vertex, y :");

scanf("%d",&yp[1]);

printf("\nEnter the dimension of z-axis : ");

scanf("%d",&z);

clrscr();

cleardevice();

display();

rotation();

getch();

closegraph();

}

**/\***

**Name : Rohit Aggarwal**

**Roll No. : 7CS-097**

**\*/**