import java.util.\*;

import java.awt.\*;

import java.applet.\*;

import java.math.\*;

/\*

<applet code="T3DA.class" width=500 height=500>

</applet>

\*/

public class T3DA extends Applet

{

static int A[][]=new int[100][4];

static int C[][]=new int[100][4];

static int B[][]=new int[4][4];

static double draw1[][]=new double[100][2];

static double draw2[][]=new double[100][2];

static int n=0;

public void init()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the no. of co-ordinates");

n=sc.nextInt();

for(int i=0; i<n; i++)

{

System.out.println("Enter the x-co-ordinates of point " +(i+1));

A[i][0]=sc.nextInt();

System.out.println("Enter the y-co-ordinates of point " +(i+1));

A[i][1]=sc.nextInt();

System.out.println("Enter the z-co-ordinates of point " +(i+1));

A[i][2]=sc.nextInt();

A[i][3]=1;

}

A[n][2]=1;

System.out.println("Enter the x-translation unit");

B[3][0]=sc.nextInt();

System.out.println("Enter the y-translation unit");

B[3][1]=sc.nextInt();

System.out.println("Enter the z-translation unit");

B[3][2]=sc.nextInt();

B[2][2]=1;

B[0][0]=1;

B[1][1]=1;

B[3][3]=1;

for(int i=0;i<n;i++)

{

draw1[i][0]=A[i][0]+A[i][2]\*Math.cos(Math.toRadians(60));

draw1[i][1]=A[i][1]+A[i][2]\*Math.sin(Math.toRadians(60));

}

for(int j=0; j<n+1; j++)

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

for(int k=0; k<4;k++)

C[j][i]+=A[j][k]\*B[k][i];

for(int i=0;i<n;i++)

{

draw2[i][0]=C[i][0]+C[i][2]\*Math.cos(Math.toRadians(60));

draw2[i][1]=C[i][1]+C[i][2]\*Math.sin(Math.toRadians(60));

}

}

public void paint(Graphics g)

{

for(int i=0; i<n-1; i++)

{

g.drawLine((int)draw1[i][0], (int)draw1[i][1], (int)draw1[i+1][0], (int)draw1[i+1][1]);

g.drawLine((int)draw2[i][0], (int)draw2[i][1], (int)draw2[i+1][0], (int)draw2[i+1][1]);

}

if(n==8)

{

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

{

g.drawLine((int)draw1[i][0], (int)draw1[i][1], (int)draw1[7-i][0], (int)draw1[7-i][1]);

g.drawLine((int)draw2[i][0], (int)draw2[i][1], (int)draw2[7-i][0], (int)draw2[7-i][1]);

}

g.drawLine((int)draw1[0][0], (int)draw1[0][1], (int)draw1[3][0], (int)draw1[3][1]);

g.drawLine((int)draw1[4][0], (int)draw1[4][1], (int)draw1[7][0], (int)draw1[7][1]);

g.drawLine((int)draw2[0][0], (int)draw2[0][1], (int)draw2[3][0], (int)draw2[3][1]);

g.drawLine((int)draw2[4][0], (int)draw2[4][1], (int)draw2[7][0], (int)draw2[7][1]);

}

}

}

/\*

Output: D:\>javac T3DA.java

D:\>appletviewer T3DA.java

Enter the no. of co-ordinates

8

Enter the x-co-ordinates of point 1

12

Enter the y-co-ordinates of point 1

12

Enter the z-co-ordinates of point 1

12

Enter the x-co-ordinates of point 2

12

Enter the y-co-ordinates of point 2

100

Enter the z-co-ordinates of point 2

12

Enter the x-co-ordinates of point 3

100

Enter the y-co-ordinates of point 3

100

Enter the z-co-ordinates of point 3

12

Enter the x-co-ordinates of point 4

100

Enter the y-co-ordinates of point 4

12

Enter the z-co-ordinates of point 4

12

Enter the x-co-ordinates of point 5

100

Enter the y-co-ordinates of point 5

12

Enter the z-co-ordinates of point 5

100

Enter the x-co-ordinates of point 6

100

Enter the y-co-ordinates of point 6

100

Enter the z-co-ordinates of point 6

100

Enter the x-co-ordinates of point 7

12

Enter the y-co-ordinates of point 7

100

Enter the z-co-ordinates of point 7

100

Enter the x-co-ordinates of point 8

12

Enter the y-co-ordinates of point 8

12

Enter the z-co-ordinates of point 8

100

Enter the x-translation unit

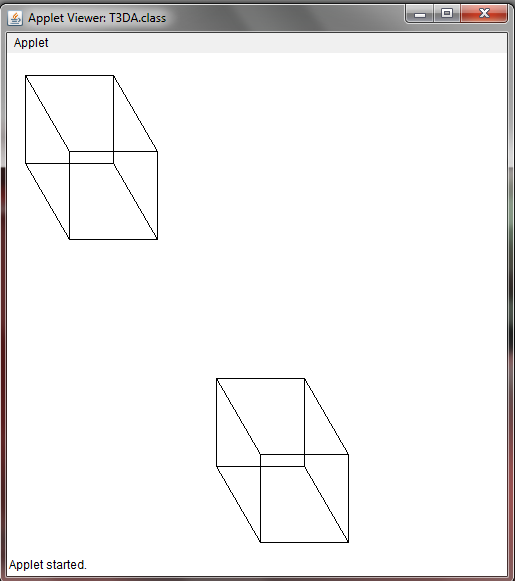
120

Enter the y-translation unit

179

Enter the z-translation unit

143



\*/