Name: Vinayak Madan Shete

Roll No.: TYCOC303

Div: C Batch: C4

Course Name: PBL3- Computer Graphics & Gaming

Course Code: BCE5504

==========

Problem Definition:

Write C++/java program to draw a house and perform following basic transformations.

i)Scaling ii) Translation iii) Rotation iv) Reflection v) Shearing

==========

Input:

```
#include<stdio.h>
#include<stdlib.h>
#include<graphics.h>
#include<math.h>


void house(int a[20][20],int edges){
    int i;


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

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

    // draw the polygon
    for (i = 0; i < edges-1; i++)
    {
        line(320+a[i][0], 240-a[i][1],
320+a[i+1][0], 240-a[i+1][1]);
    }
    line(320+a[0][0], 240-a[0][1],
320+a[i][0], 240-a[i][1]);
    line(320+a[1][0],240-
a[1][1],320+a[3][0],240-a[3][1]);
}



void transformation(int a[20][20],int
tmatrix[20][20],int edges,int code){
    int result[20][20];
    int i;

    // multiply a with tmatrix and
store in result
    for (i = 0; i < edges; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            result[i][j] = 0;
```

```
            for (int k = 0; k < 3; k++)
            {
                result[i][j] += a[i][k]
* tmatrix[k][j];
            }
        }
    }

    // draw the polygon
    setcolor(code);
    for (i = 0; i < edges-1; i++)
    {
        line(320+result[i][0], 240-
result[i][1], 320+result[i+1][0], 240-
result[i+1][1]);
    }
    line(320+result[0][0], 240-
result[0][1], 320+result[i][0], 240-
result[i][1]);
    line(320+result[1][0],240-
result[1][1],320+result[3][0],240-
result[3][1]);
}



int main()
{
 int gd = DETECT, gm;
 initgraph(&gd, &gm, NULL);

    int
a[20][20],tmatrix[20][20],result[20][20
];
    int edges,conti,code;
    int choice;
    float radian;
    printf("\nEnter the number of edges
to draw a House: ");
    scanf("%d", &edges);
    for (int i = 0; i < edges; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            if(j>=2){
                a[i][j] = 1;
            }
            else{
                scanf("%d", &a[i][j]);
            }
        }
    }
    house(a,edges);

    setcolor(RED);
    outtextxy(450, 20, "Translated
Polygon");
    setcolor(GREEN);
    outtextxy(450, 40, "Rotated
Polygon");
    setcolor(CYAN);
    outtextxy(450, 60, "Scaled
Polygon");
    setcolor(YELLOW);
    outtextxy(450, 80, "Reflected
Polygon");
    setcolor(MAGENTA);
    outtextxy(450, 100, "Sheared
Polygon");

    printf("\nEnter your choice:");

do
{
     printf("\nSelection the
Transformation you want to apply ==>
\n1. Translate\n2. Rotate\n3. Scale\n4.
Reflection\n5. Shear\n6. Exit\nEnter
your choice ==>");
    scanf("%d", &choice);
```

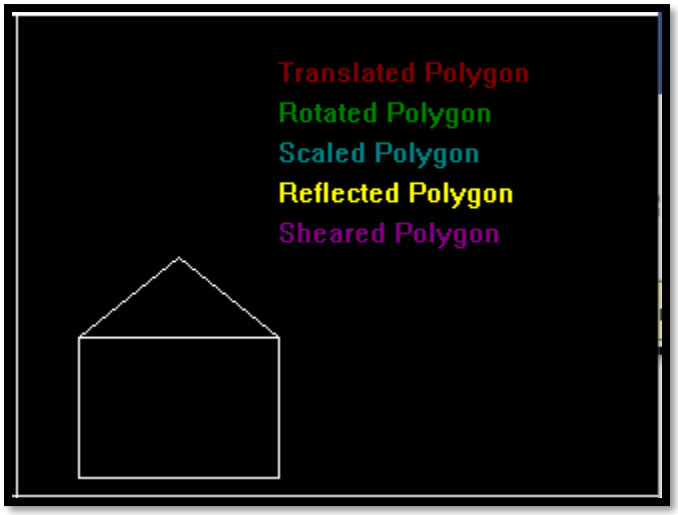
```
    switch (choice)
    {
    case 1:
        int i,tx,ty;
        printf("\nEnter the translation factor Tx and Ty => ");
        scanf("%d %d", &tx, &ty);

        tmatrix[0][0] = tmatrix[1][1] = tmatrix[2][2] = 1;
        tmatrix[0][1] = tmatrix[0][2] = tmatrix[1][0] = tmatrix[1][2] = 0;
        tmatrix[2][0] = tx;
        tmatrix[2][1] = ty;

transformation(a,tmatrix,edges,4);

        break;

    case 2:
        int angle;
        printf("\nEnter the angle of rotation => ");
        scanf("%d", &angle);

           // Convert the angle to radians
//        radian = (angle * 3.14159265359) / 180.0;
radian = angle;

    tmatrix[0][0] = cos(radian);
    tmatrix[0][1] = -sin(radian);
    tmatrix[1][0] = sin(radian);
    tmatrix[1][1] = cos(radian);
    tmatrix[2][2] = 1;
    tmatrix[0][2] = tmatrix[1][2] = tmatrix[2][0] = tmatrix[2][1] = 0;

    transformation(a,tmatrix,edges,2);
    break;

    case 3:
        int sx,sy;
        setcolor(YELLOW);
        printf("\nEnter the scaling factor Sx and Sy => ");
        scanf("%d %d", &sx, &sy);

        tmatrix[0][0] = sx;
        tmatrix[1][1] = sy;
        tmatrix[2][2] = 1;
        tmatrix[0][1] = tmatrix[0][2] = tmatrix[1][0] = tmatrix[1][2] = tmatrix[2][0] = tmatrix[2][1] = 0;

transformation(a,tmatrix,edges,1);
    break;

    case 4:
        int choice;
        setcolor(YELLOW);
        printf("\nSelect the type of Reflection==> \n1. About X-axis\n2. About Y-axis\n3. About Origin\n4. About X=Y\nEnter your choice =>");
        scanf("%d", &choice);

        switch (choice)
        {
        case 1:
            tmatrix[0][0] = tmatrix[2][2] = 1;
            tmatrix[1][1] = -1;
            tmatrix[0][1] = tmatrix[0][2] = tmatrix[1][0] = tmatrix[1][2] = tmatrix[2][0] = tmatrix[2][1] = 0;

transformation(a,tmatrix,edges,14);
            break;
```

```
        case 2:
            tmatrix[0][0] = -1;
            tmatrix[1][1] =
tmatrix[2][2] = 1;
            tmatrix[0][1] =
tmatrix[0][2] = tmatrix[1][0] =
tmatrix[1][2] = tmatrix[2][0] =
tmatrix[2][1] = 0;

transformation(a,tmatrix,edges,14);
            break;


        case 3:
            tmatrix[0][0] =
tmatrix[1][1] = -1;
            tmatrix[2][2] = 1;
            tmatrix[0][1] =
tmatrix[0][2] = tmatrix[1][0] =
tmatrix[1][2] = tmatrix[2][0] =
tmatrix[2][1] = 0;

transformation(a,tmatrix,edges,14);
            break;


        case 4:
            tmatrix[0][0] =
tmatrix[1][1] = 0;
            tmatrix[2][2] = 1;
            tmatrix[0][1] =
tmatrix[0][2] = tmatrix[1][0] =
tmatrix[1][2] = tmatrix[2][0] =
tmatrix[2][1] = 1;

transformation(a,tmatrix,edges,14);
            break;


        default:
            break;
        }
    break;
    case 5:
        int shx,shy;
        setcolor(MAGENTA);
        printf("\nEnter the shearing
factor Shx and Shy =>");
        scanf("%d %d", &shx, &shy);


        tmatrix[0][0] = tmatrix[1][1] =
tmatrix[2][2] = 1;
        tmatrix[0][1] = shx;
        tmatrix[1][0] = shy;
        tmatrix[0][2] = tmatrix[1][2] =
tmatrix[2][0] = tmatrix[2][1] = 0;

transformation(a,tmatrix,edges,5);
    break;


    case 6:
        exit(0);
    break;


    default:
        printf("\nInvalid choice!");
    }
    printf("\nPress 1 to continue the
Transformation => ");
    scanf("%d", &conti);


} while (conti);



 delay(500000);
 //closegraph();
 return 0;
}
```

Output:

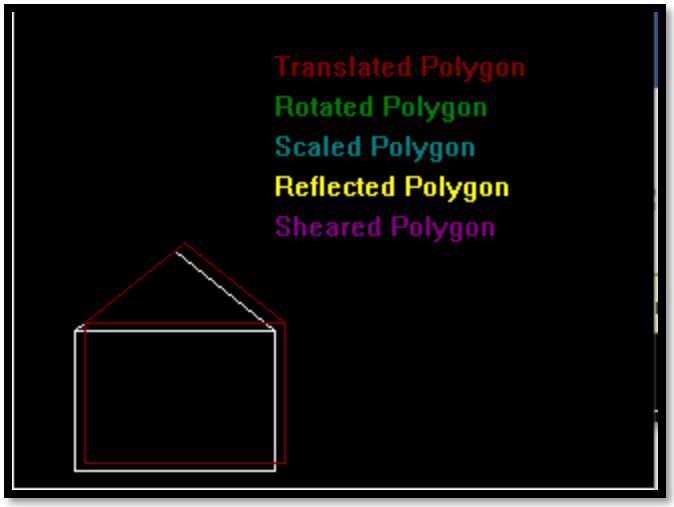
- Building a House:

```
Enter the number of edges to draw a House: 6
30 10
30 80
80 120
130 80
130 10
30 10
```



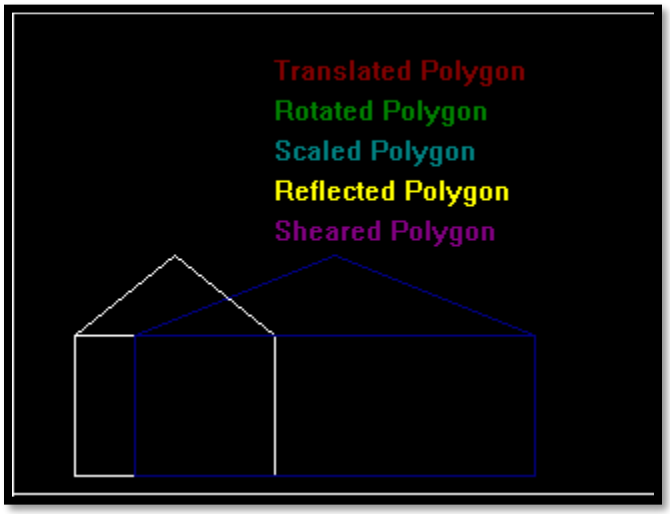
- Translation:

```
Enter your choice:
Selection the Transformation you want to apply ==>
1. Translate
2. Rotate
3. Scale
4. Reflection
5. Shear
6. Exit
Enter your choice ==>1

Enter the translation factor Tx and Ty => 5 4

Press 1 to continue the Transformation =>
```

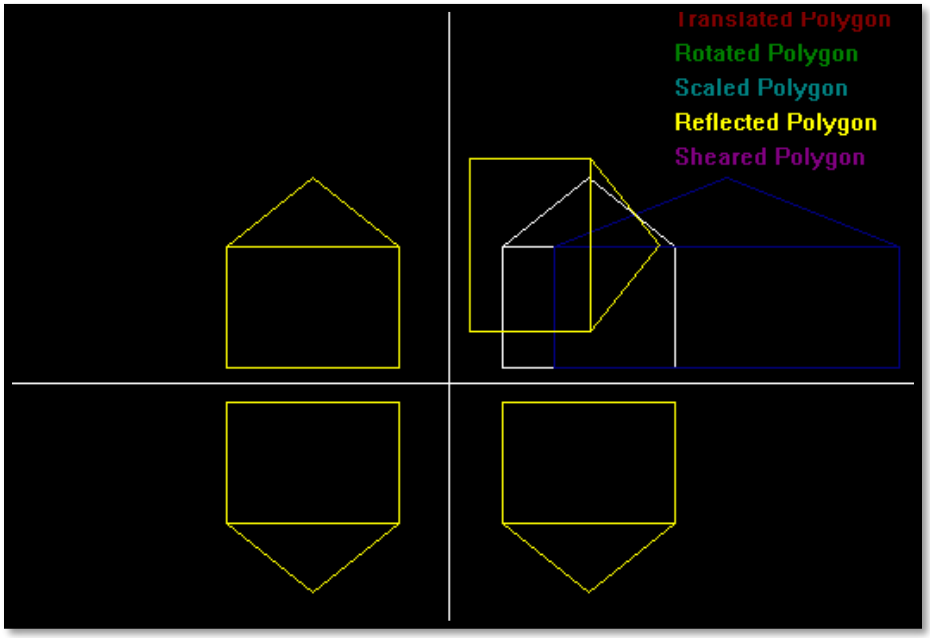


- Scaling:

```
Enter your choice:
Selection the Transformation you want to apply ==>
1. Translate
2. Rotate
3. Scale
4. Reflection
5. Shear
6. Exit
Enter your choice ==>3

Enter the scaling factor Sx and Sy => 2 1

Press 1 to continue the Transformation =>
```



- Reflection:



- Shearing:

```
Enter your choice:
Selection the Transformation you want to apply ==>
1. Translate
2. Rotate
3. Scale
4. Reflection
5. Shear
6. Exit
Enter your choice ==>5

Enter the shearing factor Shx and Shy =>2 1

Press 1 to continue the Transformation =>
```