Name: Khushi Nitinkumar Patel

PRN: 2020BTECS00037

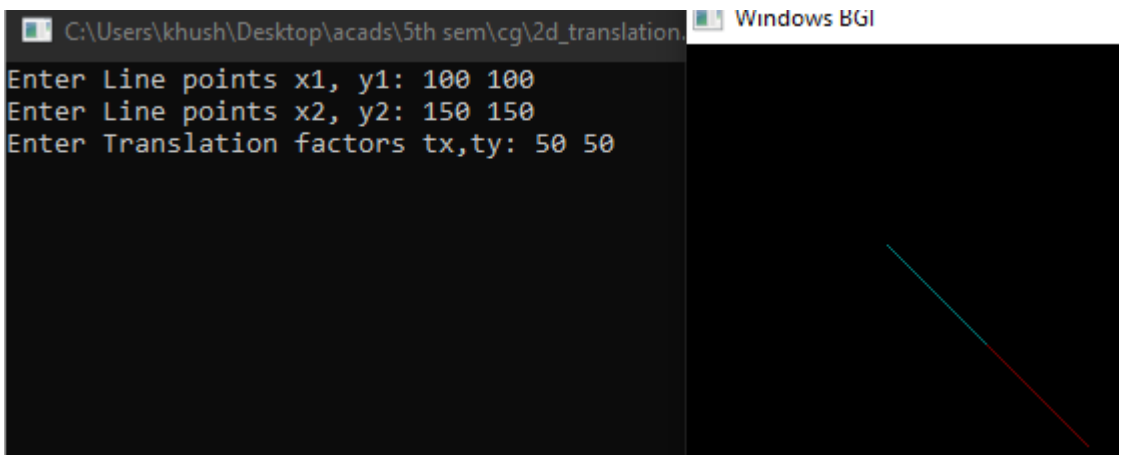
Batch: T5

Experiment 5: Implementation of 2D transformation

1.Translation

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

using namespace std;

int main(){

//TRANSLATION ON A LINE

    int gd = DETECT, gm;

    int x1, x2, y1, y2, tx, ty;

    printf("Enter Line points x1, y1: ");
    scanf("%d %d",&x1, &y1);
    printf("Enter Line points x2, y2: ");
    scanf("%d %d", &x2, &y2);
    printf("Enter Translation factors tx,ty: ");
    scanf("%d %d",&tx,&ty);
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");
    setcolor(3);
    line(x1,y1,x2,y2);

    x1 = x1+tx;
    y1 = y1+ty;
    x2 = x2+tx;
    y2 = y2+ty;

    setcolor(4);
    line(x1,y1,x2,y2);
    getch();
    closegraph();
    return 0;
}
```

OUTPUT:

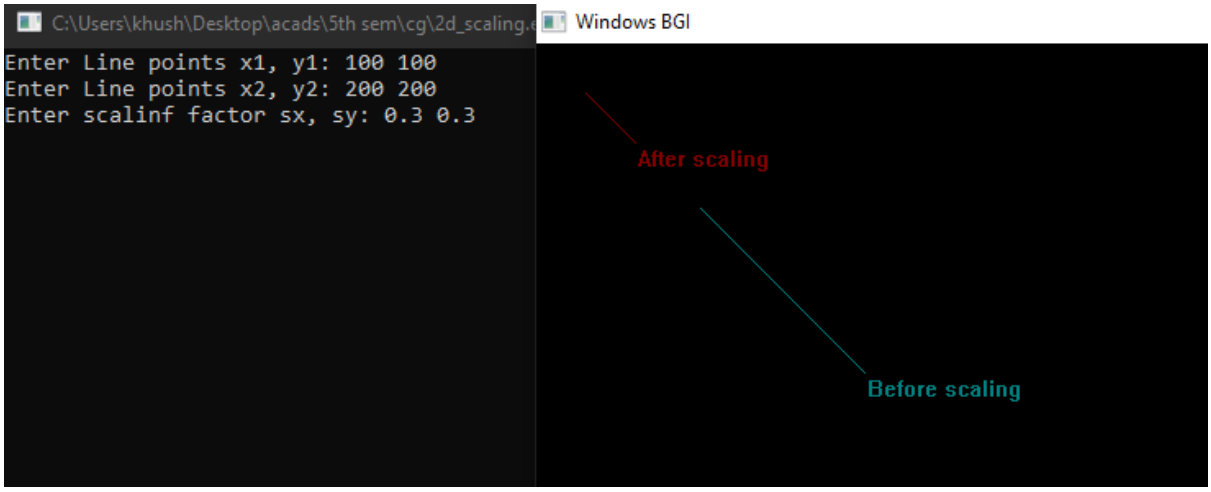


2.Scaling

2d_scaling.cpp | 2d_translation.cpp

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

using namespace std;

int main(){
    int gd = DETECT, gm;
    int x1, x2, y1, y2, x3, y3, x4, y4;
    float sx, sy;

    printf("Enter Line points x1, y1: ");
    scanf("%d %d",&x1, &y1);
    printf("Enter Line points x2, y2: ");
    scanf("%d %d", &x2, &y2);
    printf("Enter scalinf factor sx, sy: ");
    scanf("%f %f",&sx, &sy);
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");
    setcolor(3);
    line(x1,y1,x2,y2);
    outtextxy(x2+2, y2+2, "Before scaling");

    x3 = x1*sx;
    y3 = y1*sy;
    x4 = x2*sx;
    y4 = y2*sy;;
    setcolor(4);
    line(x3,y3,x4,y4);
    outtextxy(x4 + 2, y4 + 2, "After scaling");
    getch();
    closegraph();
    return 0;
}
```

OUTPUT:

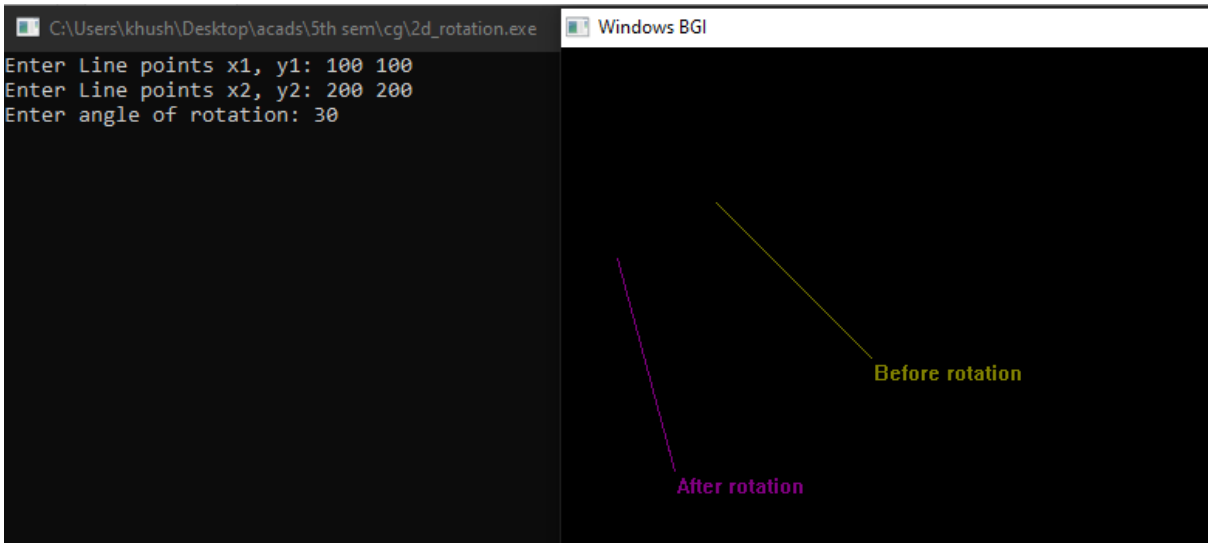


3.Rotation

2d_rotation.cpp

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

using namespace std;

int main(){


    int gd = DETECT, gm;
    int x1, x2, y1, y2, x3, y3, x4, y4;
    float angle, theta;

    printf("Enter Line points x1, y1: ");
    scanf("%d %d",&x1, &y1);
    printf("Enter Line points x2, y2: ");
    scanf("%d %d", &x2, &y2);
    printf("Enter angle of rotation: ");
    scanf("%f",&angle);
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");
    setcolor(6);
    line(x1,y1,x2,y2);
    outtextxy(x2+2, y2+2, "Before rotation");
    theta = angle * (3.14/180);

    x3 = (x1*cos(theta)) - (y1*sin(theta));
    y3 = (x1*sin(theta)) + (y1*cos(theta));
    x4 = (x2*cos(theta)) - (y2*sin(theta));
    y4 = (x2*sin(theta)) + (y2*cos(theta));
    setcolor(5);
    line(x3, y3, x4, y4);
    outtextxy(x4+2, y4+2, "After rotation");
    getch();
    closegraph();
    return 0;
}
```

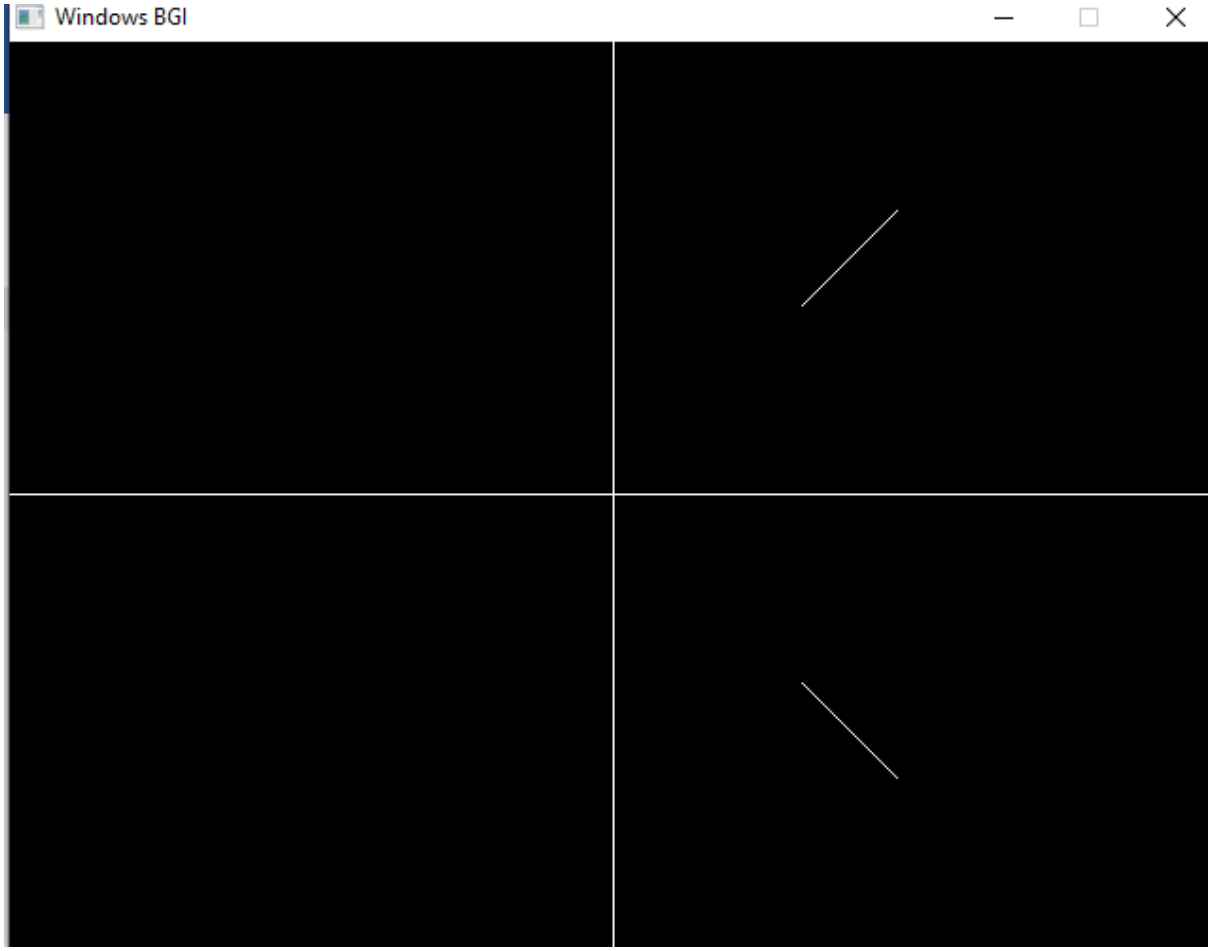
OUTPUT:



4.Reflection

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

using namespace std;

int main(){

    int gd = DETECT, gm;
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");
    int a = getmaxx();
    int b = getmaxy();
    int y = b/2;
    int x = a/2;
    line(x,0,x,b);
    line(0,y,a,y);
    int x1 = x+100;
    int y1 = y-100;
    int x2 = x+150;
    int y2 = y-150;
    line(x2,y2,x1,y1);
    line(x2,y+150,x1, y+100);
    getch();
    closegraph();
    return 0;
}
```

OUTPUT:



5.Shearing

2d_rotation.cpp | 2d_translation.cpp | 2d_reflection.cpp | 2d_shearing.cpp

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

using namespace std;

int main(){
    int gd = DETECT, gm;
    int arr[] = {100, 100, 300, 100, 300, 300, 100, 300, 100, 100};
    float shx;


    printf("Enter shear factor shx: ");
    scanf("%f",&shx);
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");

    drawpoly(5,arr);
    setcolor(6);
    int arr1[] = {(100+(100*shx)),100,
    (300+(100*shx)),100,
    (300+(300*shx)),300,
    (100+(300*shx)),300,
    (100+(100*shx)),100};

    drawpoly(5,arr1);
    setcolor(7);
    getch();
    closegraph();
    return 0;

}
```

OUTPUT: