

Name: Khushi Nitinkumar Patel

PRN: 2020BTECS00037

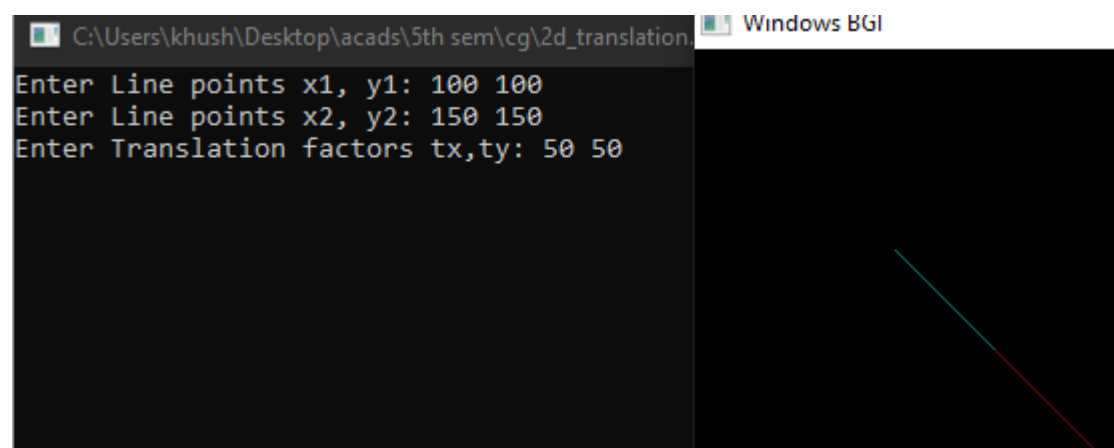
Batch: T5

Experiment 5: Implementation of 2D transformation

1.Translation

```
1  #include <stdio.h>
2  #include <graphics.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <conio.h>
6
7  using namespace std;
8
9  int main(){
10
11     //TRANSLATION ON A LINE
12
13     int gd = DETECT, gm;
14
15     int x1, x2, y1, y2, tx, ty;
16
17     printf("Enter Line points x1, y1: ");
18     scanf("%d %d",&x1, &y1);
19     printf("Enter Line points x2, y2: ");
20     scanf("%d %d", &x2, &y2);
21     printf("Enter Translation factors tx,ty: ");
22     scanf("%d %d",&tx,&ty);
23     initgraph(&gd, &gm, "c:\\\\turbo3\\bgi");
24     setcolor(3);
25     line(x1,y1,x2,y2);
26
27     x1 = x1+tx;
28     y1 = y1+ty;
29     x2 = x2+tx;
30     y2 = y2+ty;
31
32     setcolor(4);
33     line(x1,y1,x2,y2);
34     getch();
35     closegraph();
36     return 0;
37 }
```

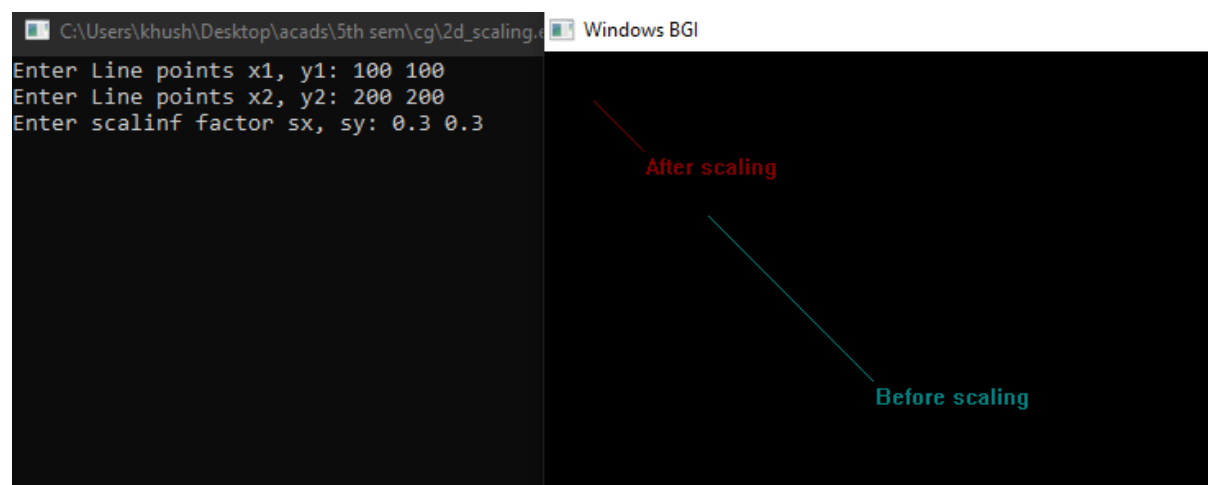
OUTPUT:



2. Scaling

```
2d_scaling.cpp 2d_translation.cpp
1  #include <stdio.h>
2  #include <graphics.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <conio.h>
6
7  using namespace std;
8
9  int main(){
10     int gd = DETECT, gm;
11     int x1, x2, y1, y2, x3, y3, x4, y4;
12     float sx, sy;
13
14     printf("Enter Line points x1, y1: ");
15     scanf("%d %d", &x1, &y1);
16     printf("Enter Line points x2, y2: ");
17     scanf("%d %d", &x2, &y2);
18     printf("Enter scaling factor sx, sy: ");
19     scanf("%f %f", &sx, &sy);
20     initgraph(&gd, &gm, "c:\\turbo3\\bgi");
21     setcolor(3);
22     line(x1, y1, x2, y2);
23     outtextxy(x2+2, y2+2, "Before scaling");
24
25     x3 = x1*sx;
26     y3 = y1*sy;
27     x4 = x2*sx;
28     y4 = y2*sy;
29     setcolor(4);
30     line(x3, y3, x4, y4);
31     outtextxy(x4 + 2, y4 + 2, "After scaling");
32     getch();
33     closegraph();
34     return 0;
35 }
```

OUTPUT:



3.Rotation

2d_rotation.cpp

```
1  #include <stdio.h>
2  #include <graphics.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <conio.h>
6
7  using namespace std;
8
9  int main(){
10
11
12     int gd = DETECT, gm;
13     int x1, x2, y1, y2, x3, y3, x4, y4;
14     float angle, theta;
15
16     printf("Enter Line points x1, y1: ");
17     scanf("%d %d", &x1, &y1);
18     printf("Enter Line points x2, y2: ");
19     scanf("%d %d", &x2, &y2);
20     printf("Enter angle of rotation: ");
21     scanf("%f", &angle);
22     initgraph(&gd, &gm, "c:\\\\turbo3\\bgi");
23     setcolor(6);
24     line(x1, y1, x2, y2);
25     outtextxy(x2+2, y2+2, "Before rotation");
26     theta = angle * (3.14/180);
27
28     x3 = (x1*cos(theta)) - (y1*sin(theta));
29     y3 = (x1*sin(theta)) + (y1*cos(theta));
30     x4 = (x2*cos(theta)) - (y2*sin(theta));
31     y4 = (x2*sin(theta)) + (y2*cos(theta));
32     setcolor(5);
33     line(x3, y3, x4, y4);
34     outtextxy(x4+2, y4+2, "After rotation");
35     getch();
36     closegraph();
37     return 0;
38 }
```

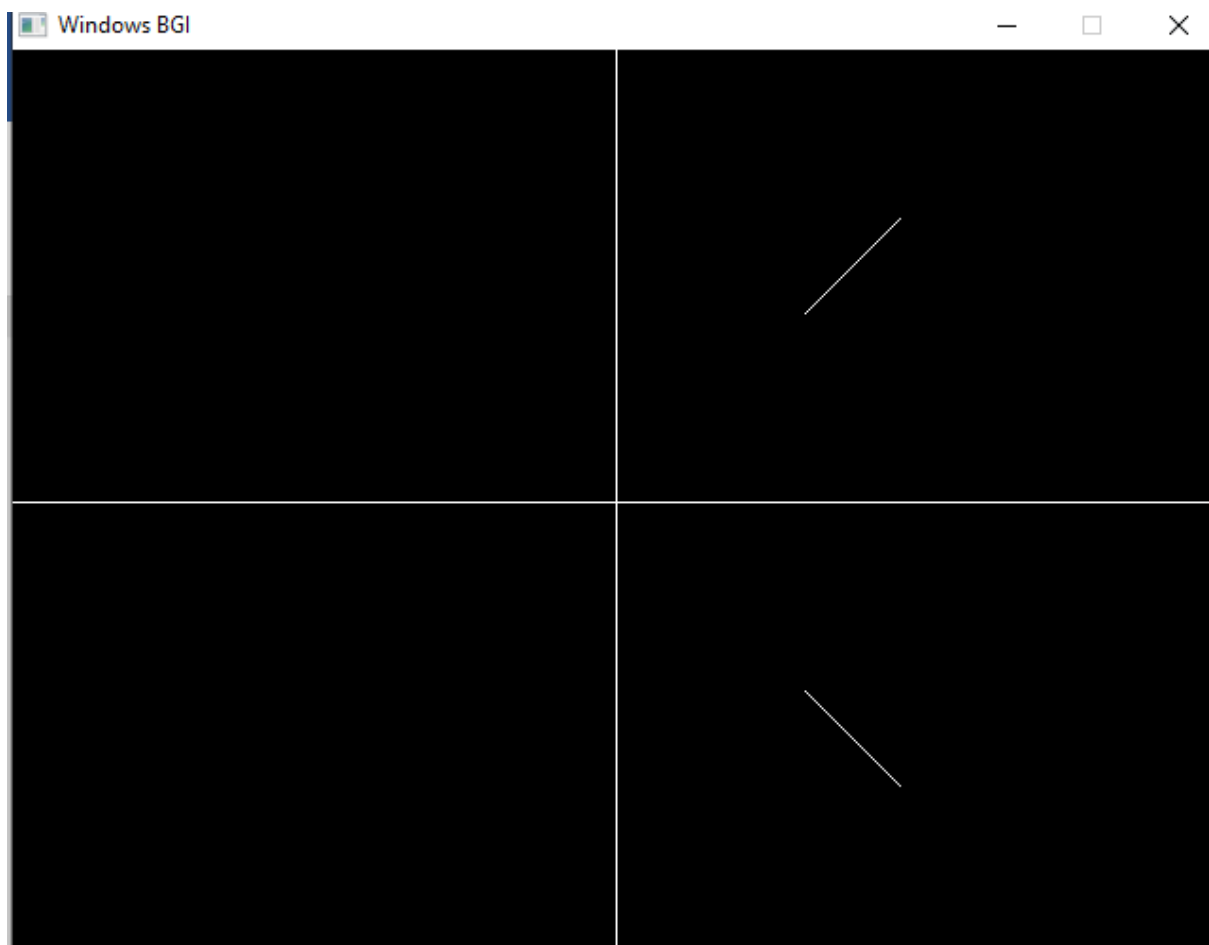
OUTPUT:



4.Reflection

```
1  #include <stdio.h>
2  #include <graphics.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <conio.h>
6
7  using namespace std;
8
9  int main(){
10
11     int gd = DETECT, gm;
12     initgraph(&gd, &gm, "c:\\turbo3\\bgi");
13     int a = getmaxx();
14     int b = getmaxy();
15     int y = b/2;
16     int x = a/2;
17     line(x,0,x,b);
18     line(0,y,a,y);
19     int x1 = x+100;
20     int y1 = y-100;
21     int x2 = x+150;
22     int y2 = y-150;
23     line(x2,y2,x1,y1);
24     line(x2,y+150,x1, y+100);
25     getch();
26     closegraph();
27     return 0;
28 }
```

OUTPUT:



5.Shearing

```
2d_rotation.cpp 2d_translation.cpp 2d_reflection.cpp 2d_shearing.cpp
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4  #include <graphics.h>
5
6  using namespace std;
7
8  int main(){
9      int gd = DETECT, gm;
10     int arr[] = {100, 100, 300, 100, 300, 300, 100, 300, 100, 100};
11     float shx;
12
13
14     printf("Enter shear factor shx: ");
15     scanf("%f",&shx);
16     initgraph(&gd, &gm, "c:\\\\turbo3\\\\bgi");
17
18     drawpoly(5,arr);
19     setcolor(6);
20     int arr1[] = {(100+(100*shx)),100,
21                 (300+(100*shx)),100,
22                 (300+(300*shx)),300,
23                 (100+(300*shx)),300,
24                 (100+(100*shx)),100};
25
26     drawpoly(5,arr1);
27     setcolor(7);
28     getch();
29     closegraph();
30     return 0;
31 }
32 }
```

OUTPUT:

