

Practical 1

Task 1 : Write a program to draw a line using the built-in line function of “Graphics.h” .

Source Code:

```
#include<graphics.h>
int main()
{
    int gd = DETECT,gm;
    initgraph(&gd,&gm,NULL);
    line(20,20,100,100);
    delay(5000);
    closegraph();
    return 0;
}
```

Output:



Task 2 : Write a program to draw a rectangle using the built-in line function of “Graphics.h” .

Source Code:

```
#include<graphics.h>
int main()
{
    int gd = DETECT, gm, left=100, top=100, right=200, bottom=200, x=
300, y=150, radius=50;
    initgraph(&gd, &gm, NULL);
    line(10, 10, 100, 10);
    line(10, 10, 10, 100);
    line(10, 100, 100, 100);
    line(100, 10, 100, 100);

    delay(5000);
    closegraph();
    return 0;
}
```

Output :



Output:

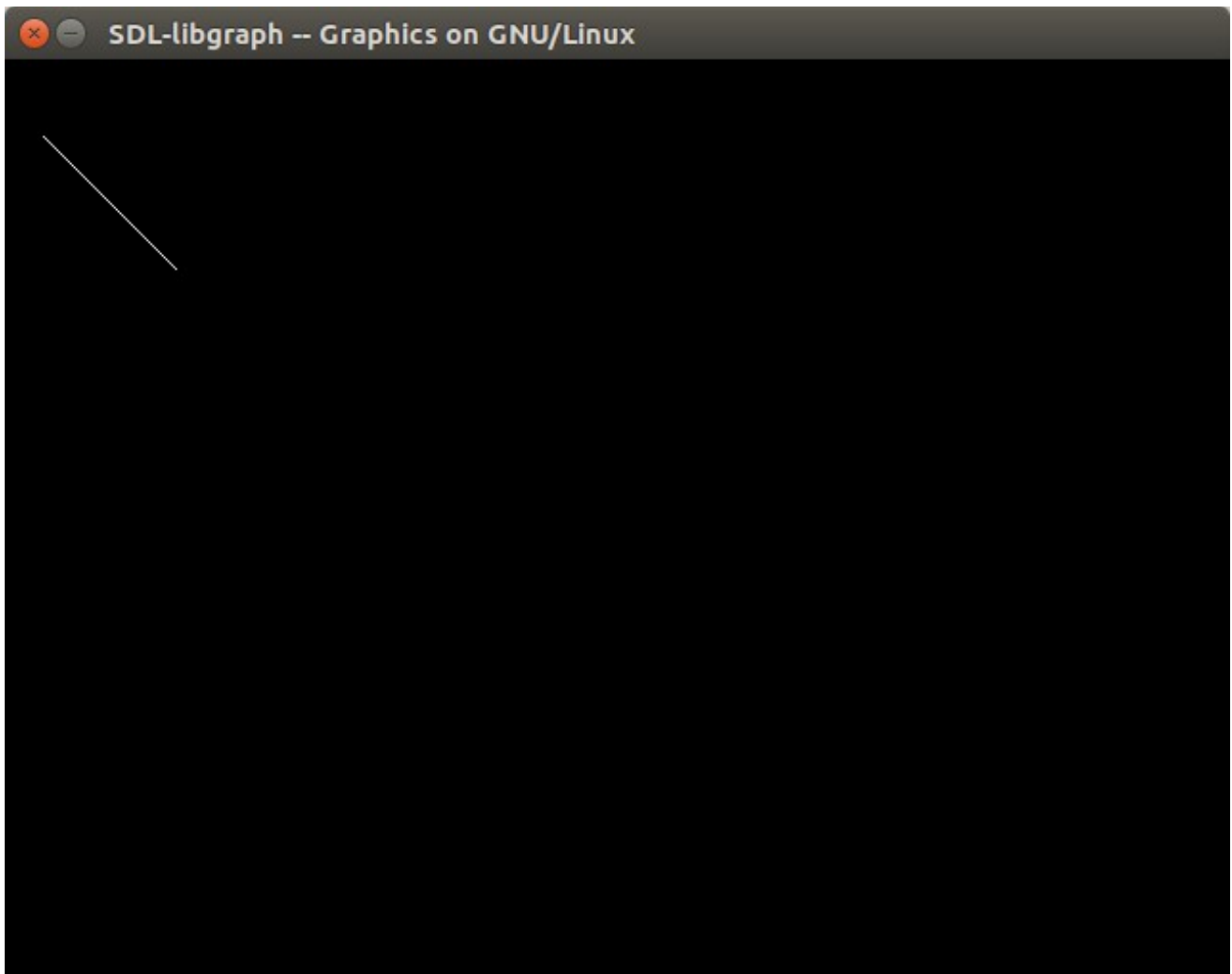
```
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ ./a.out
Enter the starting point20
30
Enter the ending point70
90
```



Output:

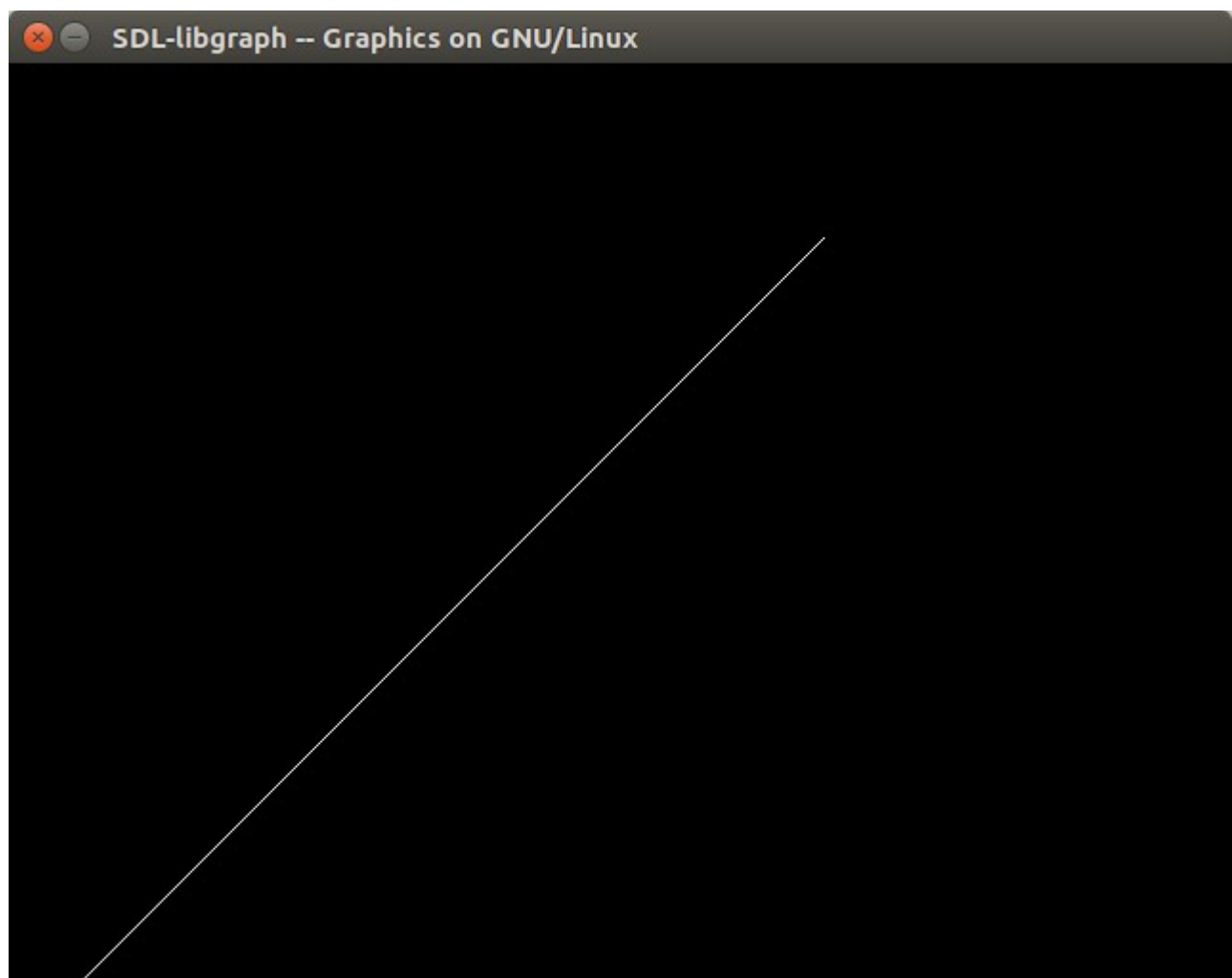
For Positive Slope:

```
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ gcc prac_3_Breshman_algo.c -lgraph
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ ./a.out
Enter the starting point20
40
Enter the ending point90
120
```



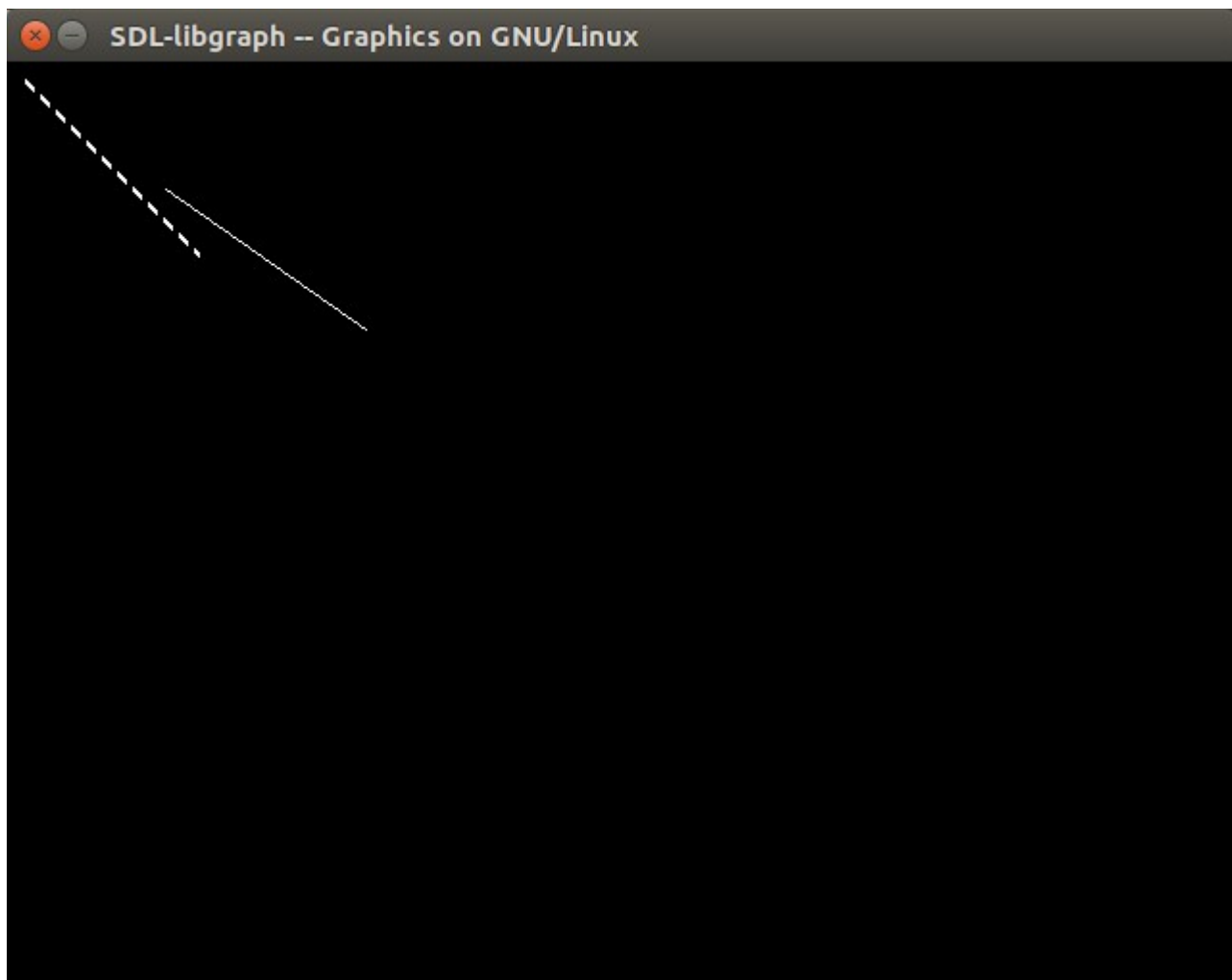
For Negative Slope:

```
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/C  
$ ./a.out  
Enter the starting point20  
500  
Enter the ending point300  
90
```



Output:

```
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ gcc prac_7_2DComposite_transformation_algo.c -lgraph -lm
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ ./a.out
Enter the starting point
10
10
Enter the ending point
100
100
Enter the type of transformation from list given
1) Translation
2) Rotation
3) Scaling
4) Shear X
5) Shear Y
6) Exit and print
1
Enter new coordinates for Translation origin
60
70
Enter the type of transformation from list given
1) Translation
2) Rotation
3) Scaling
4) Shear X
5) Shear Y
6) Exit and print
2
Enter coordinates for point about which should i rotate
1
2
Enter angle by which to rotate
-10
Enter the type of transformation from list given
1) Translation
2) Rotation
3) Scaling
4) Shear X
5) Shear Y
6) Exit and print
6
[xch] Unknown sequence number while processing queue
```

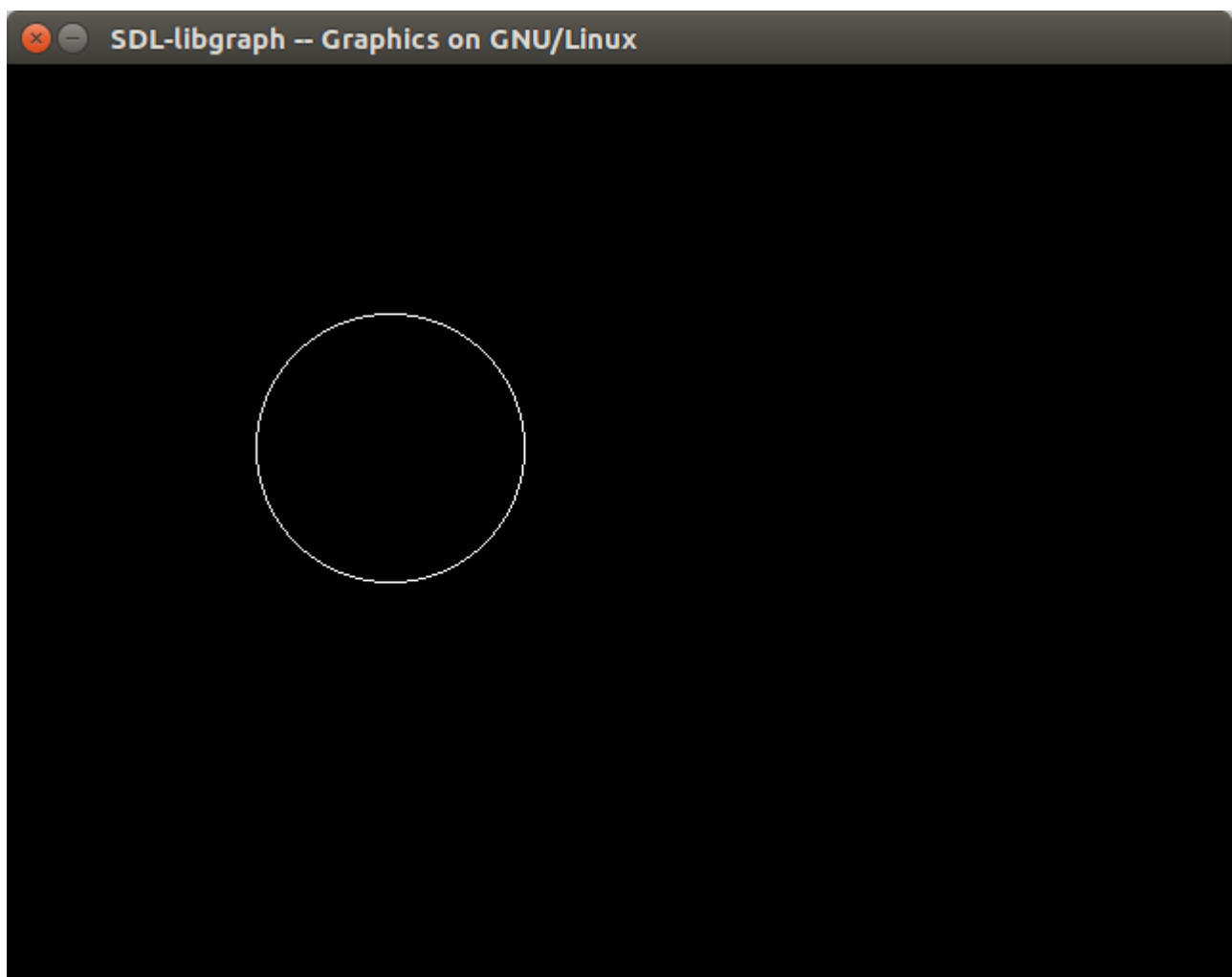


DASHED_LINE is the original line.

```
    delay(5000);  
    closegraph();  
    return 0;  
}
```

Output:

```
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG  
$ ./a.out  
Enter the Center of circle  
200  
200  
Enter the Radius of circle  
70
```



Practical 9

Task 1 : Write a program to demonstrate Bresenham's mid point ellipse algorithm.

Source Code:

```
#include<graphics.h>
#include<stdio.h>
#include<math.h>
void ellipseDrawPoints(float x_center,float y_center,float x,float y)
{
    putpixel(x_center+x,y_center+y,WHITE);
    putpixel(x_center-x,y_center+y,WHITE);
    putpixel(x_center+x,y_center-y,WHITE);
    putpixel(x_center-x,y_center-y,WHITE);
}
int main()
{
    int gd = DETECT,gm;
    float xa,ya,ra,rb;

    printf("Enter the Center of Ellipse\n");
    scanf("%f %f",&xa,&ya);
    printf("Enter the x axis length of Ellipse ' a '\n");
    scanf("%f",&ra);
    printf("Enter the y axis length of Ellipse ' b '\n");
    scanf("%f",&rb);
    initgraph(&gd,&gm,NULL);
    float x = 0,y=rb;
    float px = 0,py = 2*ra*ra*y,p=rb*rb-(ra*ra*rb)+(0.25*ra*ra);
    ellipseDrawPoints(xa,ya,x,y);
    while(px<py)
    {
        x++;
        px+=2*rb*rb;
        if(p<0)
        {
            p+=rb*rb+px;
        }
        else
        {
            y--;
```

```

        py-=2*ra*ra;
        p+=rb*rb+px-py;
    }
    ellipseDrawPoints(xa,ya,x,y);
}
p=rb*rb*(x+0.5)*(x+0.5)+ra*ra*(y-1)*(y-1)-ra*ra*rb*rb;
while(y>0)
{
    y--;
    py-=2*ra*ra;
    if(p>0)
    {
        p+=ra*ra-py;
    }
    else
    {
        x++;
        px+=2*rb*rb;
        p+=ra*ra+px-py;
    }
    ellipseDrawPoints(xa,ya,x,y);
}

delay(5000);
closegraph();
return 0;
}

```

Output:

```
$ gcc prac_9_midpointellipse_algo.c -lgraph
adnrs96@aditya-hp-envy-15-notebook-pc:/media/adnrs96/Local Disk/Local Disk(G)/CG
$ ./a.out
Enter the Center of Ellipse
300
250
Enter the x axis length of Ellipse ' a '
110
Enter the y axis length of Ellipse ' b '
40
```

